Uvod u kriptografiju

Grupe

Grupa je skup G zajedno sa operacijom *, tako da važe sledeća svojstva:

1. Zatvorenost: ako su a i b elementi skupa G, onda je i a * b element skupa G
2. Asocijativnost: za sve a, b i c iz G važi (a * b) * c = a * (b * c)
3. Neutralni element: postoji element e iz G takav da za svako a iz G važi e * a = a * e = a
4. Inverzni element: za svako a iz G postoji element a^{-1} iz G takav da važi: a * a^{-1} = a^{-1} * a = e

Primeri grupa

Primer 1: $(N_0, +)$ nije grupa
Zatvorenost: DA
Asocijativnost: DA
Neutralni element: DA, to je 0
Inverzni element: NE

Primer 2: (Z, +) je grupa
Zatvorenost: DA
Asocijativnost: DA
Neutralni element: DA, to je 0
Inverzni element: DA, za a je inverz -a

Primer 3: (Z, ·) nije grupa
Zatvorenost: DA
Asocijativnost: DA
Neutralni element: DA, to je 1
Inverzni element: NE

Primer 4: (Q\{0}, ·) je grupa
Zatvorenost: DA
Asocijativnost: DA
Neutralni element: DA, to je 1
Inverzni element: DA, za a je inverz 1/a

Ciklična grupa Z_p^*

Neka je p prost broj. Z_p^* je skup {1, 2, ..., p - 1}. Operacija u ovoj grupi je množenje po modulu p, definisana sa:

$a \bullet b = (a * b) \bmod p$

Ova struktura je grupa jer važi:

- rezultat množenja opet pripada Zp*
- operacija je asocijativna
- neutralni element je 1
- svaki element ima inverz po modulu p

Primer: $Z_{11}^* = \{1, 2, 3, ..., 10\}$

Posmatrajmo stepene broja 2 po modulu 11:

2, 4, 8, 5, 10, 9, 7, 3, 6, 1

Dobili smo sve elemente iz Z_{11}^*.

Zato je 2 generator grupe Z_{11}^*.

Generator grupe

Neka je G grupa.
Element g iz G nazivamo generatorom grupe G ako se pomoću njegovih stepena mogu dobiti svi elementi grupe.

Ako grupa ima bar jedan generator, kažemo da je grupa ciklična.
Za prost broj p, grupa Z_p^* je uvek ciklična.

Da li postoji brži način da proverimo da li je neki element generator?

Provera generatora u Zp*

Neka je p prost broj i neka je g element grupe Z_p^*.
Element g je generator grupe Z_p^* ako za svaki prost delilac q broja p - 1 važi: gˆ((p - 1) / q) ≠ 1 mod p.

Primer: Z_{11}^*
p - 1 = 10 = 2 * 5
Prosti delioci broja 10 su 2 i 5.

Proveravamo da li je 2 generator:
2^5 mod 11 = 10 ≠ 1
2^2 mod 11 = 4 ≠ 1
Zato je 2 generator grupe Z_{11}^*.

Proveravamo da li je 3 generator:
3^5 mod 11 = 1
Zato 3 nije generator grupe Z_{11}^*.

Generatori grupe Z_{11}^* su: 2, 6, 7 i 8

Problem diskretnog logaritma (Discrete logarithm problem)

Dato je:
p – prost broj
g – generator grupe Z_p^*
$b \in Z_p^*$

Treba pronaći x takvo da:
$g^x \equiv b \bmod p$
odnosno:
x = log_g b

Problem je lak za proveru, ali težak za rešavanje kada su brojevi veliki.
Ako znamo x, lako računamo $g^x \bmod p$.
Ako znamo samo g, p i b, teško je pronaći x.

Primer: Element g = 7 je generator grupe Z_{17}^*.
Odrediti $\log_7 8$ u grupi $Z_{17}^* = \{1, 2, \ldots, 16\}$

Računamo stepene broja 7 po modulu 17 i dobijamo:
$7^{14} \bmod 17 = 8$

Dakle: $\log_7 8 = 14$

Ciklična grupa Z_n^*

Opštije, postoji i Z_n^*, gde n ne mora biti prost broj. Tada u grupu ne ulaze svi nenulti elementi, već samo oni koji su uzajamno prosti sa n.

Ta grupa se najviše koristi u RSA kriptografiji.

Konačno polje F_p

Polje je skup u kome možemo da sabiramo, oduzimamo, množimo i delimo nenultim elementima.

U kriptografiji koristimo konačno polje:
$F_p=\{0, 1, 2, \dots, p-1\}$, gde je p prost broj.

Sve operacije u F_p računaju se po modulu p.

Važno je razlikovati F_p kao polje i $F_p^*=F_p\setminus\{0\}$ kao multiplikativnu grupu.

Eliptičke krive

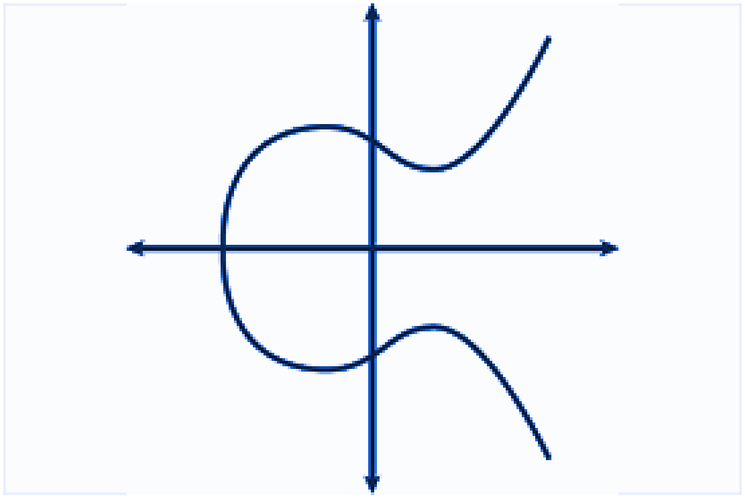
Eliptička kriva je skup tačaka koje zadovoljavaju jednačinu oblika:

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

gde su a i b elementi nekog polja.

Najčešće posmatramo eliptičke krive nad konačnim poljem F_p={0, 1, 2, ..., p - 1}, gde je p veliki prost broj. Sabiranje i množenje se rade po modulu p.

Skup tačaka na eliptičkoj krivi, zajedno sa posebnom tačkom O, formira grupu. Tačka O se naziva tačkom u beskonačnosti i ima ulogu neutralnog elementa.



Primer

Posmatrajmo eliptičku krivu nad poljem F_{11}:

$$E/F_{11} : y^2 = x^3 + 4x + 3$$

Njene tačke su parovi (x, y) iz F_{11} koji zadovoljavaju jednačinu krive:
(0, 5), (0, 6),
(3, 3), (3, 8),
(5, 4), (5, 7),
(6, 1), (6, 10),
(7, 0)
(9, 3), (9, 8),
(10, 3), (10, 8),
O (tačka u beskonačnosti).

Ukupno ima 14 tačaka.

Sabiranje tačaka

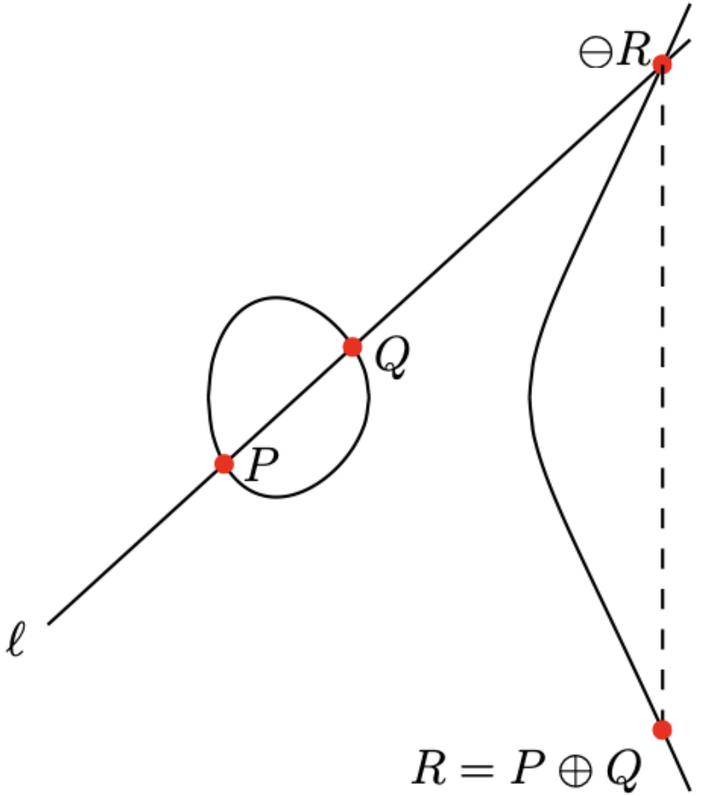
Ako imamo dve tačke P i Q:

1. povučemo pravu kroz P i Q
2. ta prava seče krivu u još jednoj tački
3. tu tačku preslikamo preko x-ose
4. dobijena tačka R predstavlja zbir tačaka P i Q.

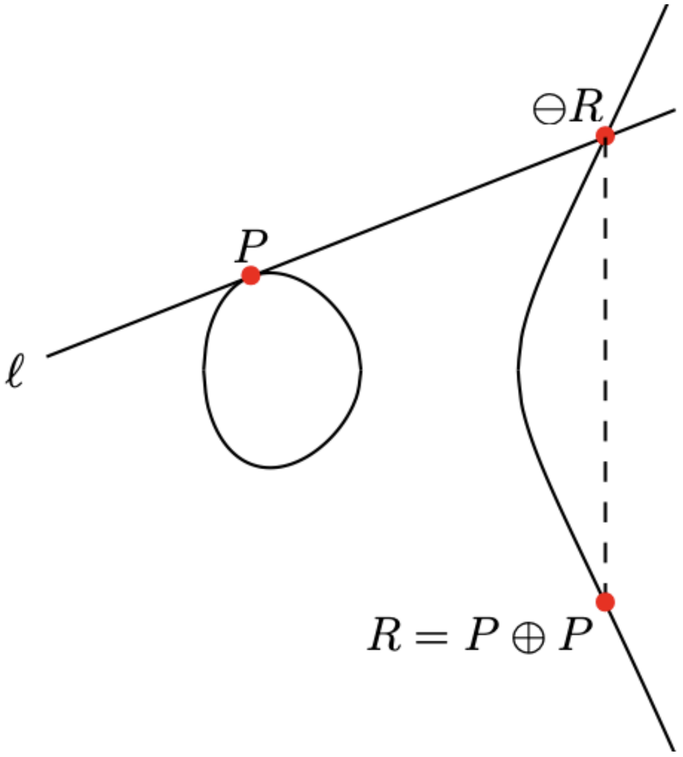
Ako su P i Q iste tačke, koristi se tangenta u tački P.

Ovako definisano sabiranje tačaka ćemo najčešće označavati sa $\oplus$.

Chord rule



Tangent rule



Eliptička kriva kao grupa

Skup tačaka na eliptičkoj krivi, zajedno sa operacijom sabiranja $\oplus$, formira Abelovu grupu, odnosno važe zatvorenost, asocijativnost, komutativnost i postoje neutralni element i inverzni element.

Neutralni element je tačka O, odnosno tačka u beskonačnosti.
Ako je $P = (x, y)$, onda je njen inverz tačka $-P = (x, -y)$.

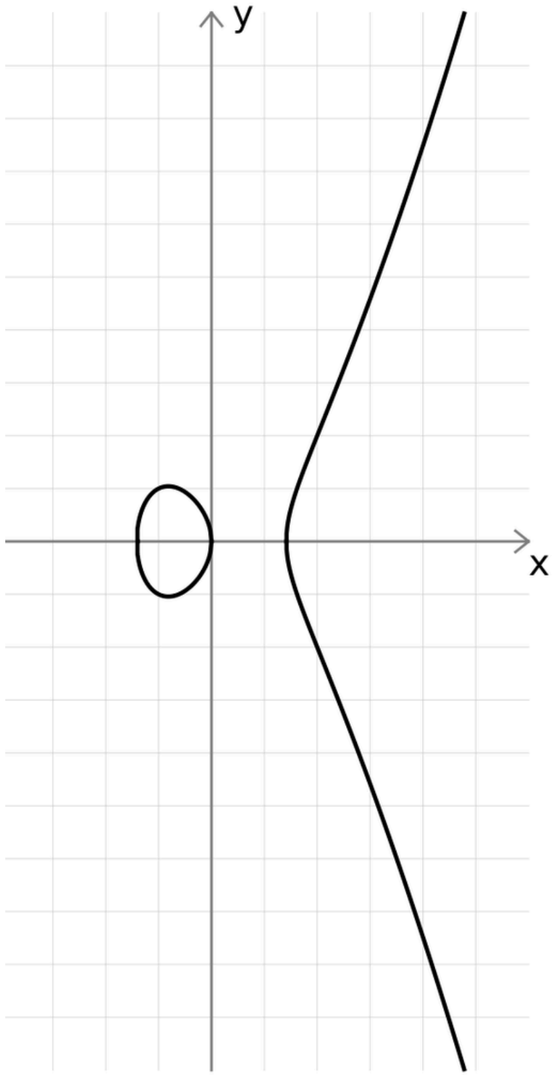
Eliptička kriva nad poljem realnih brojeva

$$E/\mathbb{R} : y^2 = x^3 - 2x$$

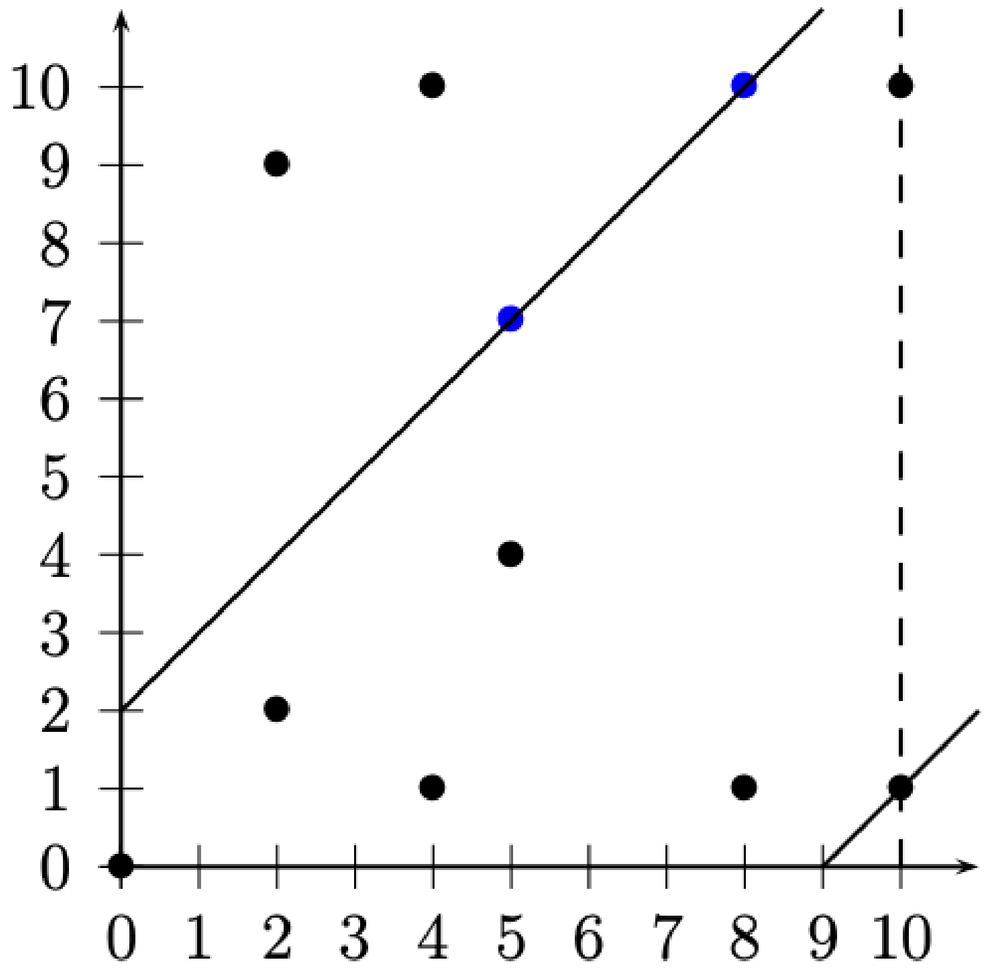
Primetimo da tačke P(-1,-1) i Q(0,0) pripadaju krivoj E.
Hoćemo da nađemo R, tako da je R=P⊕Q.

Prava koja sadrži tačke P i Q ima jednačinu: y=x

Presek te prave i eliptičke krive je tačka (2,2).
Dakle, P⊕Q=R(2,-2).



Eliptička kriva nad poljem F_{11}



$$E/\mathrm{F}_{11} : y^2 = x^3 - 2x$$

Primetimo da tačke P(5,7) i Q(8,10) pripadaju krivoj E/F_{11}.
Hoćemo da nađemo tačku R=P⊕Q.

Prava koja prolazi kroz tačke P i Q ima jednačinu y=x+2.

Presek te prave i krive je tačka (10,1),
pa je R tačka sa koordinatama (10,-1), odnosno (10,10).

Formula za sabiranje različitih tačaka

Sabiramo tačke $P(x_1, y_1)$ i $Q(x_2, y_2)$, gde je $x_1 \neq x_2$, na eliptičkoj krivoj $y^2 = x^3 + ax + b$.

Koeficijent pravca prave kroz tačke P i Q je $\lambda = (y_2 - y_1) / (x_2 - x_1)$.

Izjednačavanjem jednačine prave i eliptičke krive dobijamo x-koordinatu treće presečne tačke: $x_3 = \lambda^2 - x_1 - x_2$

Zatim dobijamo i y-koordinatu: $y_3 = \lambda(x_3 - x_1) + y_1$

Rezultujuću tačku dobijamo simetrijom u odnosu na x-osu:
$R = P \oplus Q = (x_3, -y_3)$

Eliptička kriva nad poljem F_{23}

$$E/\mathbb{F}_{23} : y^2 = x^3 + 5x + 7$$

Primetimo da tačka P(2,5) pripada krivoj E/F_{23}.

Pronaći R, tako da R=P⊕P.

$$y(x) = \sqrt{x^3 + 5x + 7} \qquad y'(x) = \frac{1}{2} \cdot \frac{3x^2 + 5}{\sqrt{x^3 + 5x + 7}}$$

$$k = y'(2) = 17 * 10^{-1} = 17 * 7 = 119 = 4 \,(\mathrm{mod}\, 23)$$

$$l : y = 4x + 20 \qquad x_R = 12$$

$$R(12, 1)$$

Formula za dupliranje tačke

Dupliramo tačku $P(x_1, y_1)$ na eliptičkoj krivoj $y^2 = x^3 + ax + b$.

Koeficijent pravca tangente na krivu u tački P je $\lambda = (3x_1^2 + a) / (2y_1)$.

Izjednačavanjem jednačine tangente i eliptičke krive dobijamo x-koordinatu presečne tačke: $x_3 = \lambda^2 - 2x_1$.

Zatim dobijamo i y-koordinatu rezultujuće tačke: $y_3 = \lambda(x_3 - x_1) + y_1$.

Rezultujuću tačku dobijamo simetrijom u odnosu na x-osu:
$R = P \oplus P = (x_3, -y_3)$

Double-and-add algoritam

Tačku P možemo da pomnožimo skalarom m tako što izvršimo m–1 sabiranja:
$m*P = P \oplus P \oplus P \oplus P \oplus \ldots \oplus P$

Ali ovaj metod je previše spor za veliko m!

Za brže računanje broj m zapisujemo u binarnom obliku i rezultat dobijamo u logaritamskom vremenu.

Na primer, da bismo izračunali 79*P, prvo broj 79 pretvaramo u binarni zapis.
Zatim možemo izračunati zbir:
$79*P = 2^6*P \oplus 2^3*P \oplus 2^2*P \oplus 2^1*P \oplus 2^0*P$.

Multi-Scalar Multiplication

Računski najzahtevniji deo u algoritmu za generisanje dokaza kod većine ZK sistema zasnovanih na eliptičkim krivama jeste algoritam za množenje tačaka skalarom, odnosno Multi-Scalar Multiplication (MSM).

Naivni algoritam koristi double-and-add algoritam.
Najbrži pristup je varijanta Pippenger-ovog algoritma, koja se naziva bucket method.

Problem diskretnog logaritma na eliptičkoj krivoj

Neka je P tačka na eliptičkoj krivoj i neka je Q = m*P.

Ako su nam poznate tačke P i Q, problem pronalaženja broja m naziva se problem diskretnog logaritma na eliptičkim krivama, odnosno Elliptic Curve Discrete Logarithm Problem (ECDLP).

Dakle, lako je izračunati Q = m*P, ali je veoma teško iz poznatih P i Q odrediti m.

Sigurnost

Grupe eliptičkih krivih relativno male veličine pružaju isti nivo sigurnosti kao multiplikativne grupe nad mnogo većim konačnim poljima.

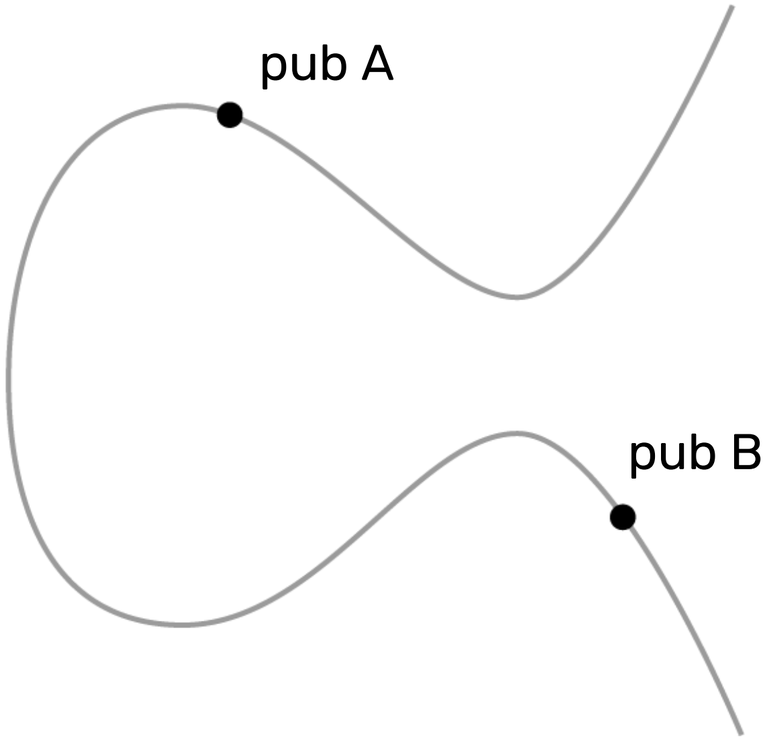
Na primer, eliptička kriva nad poljem veličine 160 bita daje nivo sigurnosti uporediv sa klasičnim sistemom zasnovanim na diskretnom logaritmu u konačnom polju veličine 1248 bita.

Privatni (private) i javni (public) ključ

Neka je G generator tačka na eliptičkoj krivoj nad konačnim poljem F_p, gde je p prost broj.

Neka su prA, prB $\in F_p$ privatni ključevi dve različite osobe, Alise i Bobana.

Tada su
pubA = prA * G
pubB = prB * G
njihovi javni ključevi.



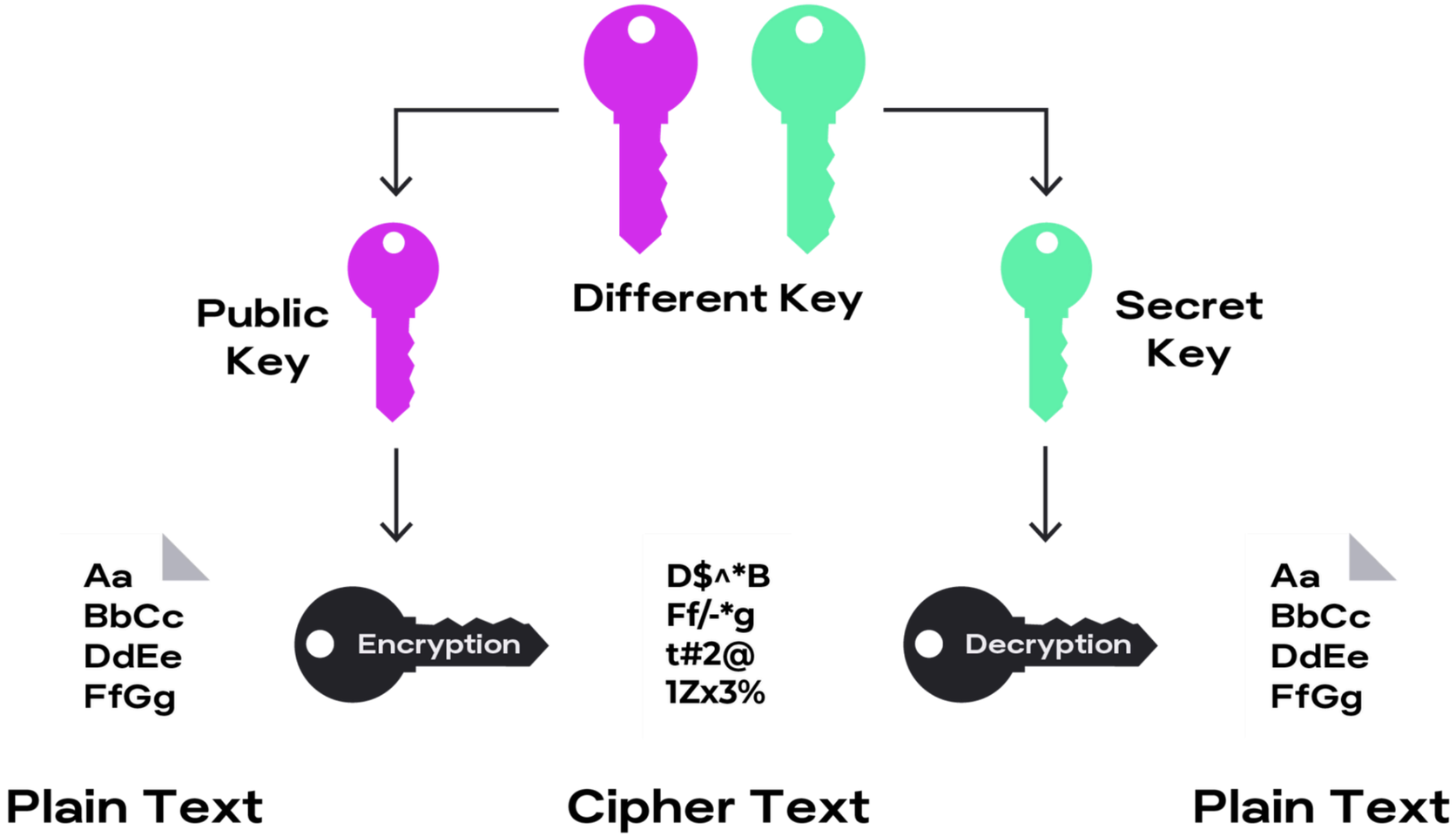
Diffie–Hellman razmena ključa

Cilj je da dve osobe naprave zajednički tajni ključ preko javnog kanala.
Alisa bira privatni ključ a i objavljuje svoj javni ključ A=a*G.
Boban bira privatni ključ b i objavljuje svoj javni ključ B=b*G.

Alisa i Boban mogu da izračunaju zajednički tajni ključ:
Alisa računa a*B=a*(b*G)=(ab)*G
Boban računa: b*A=b*(a*G)=(ab)*G

Napadač vidi G, A i B, ali zbog problema diskretnog logaritma ne može lako da izračuna a i b, a samim tim ni Bobanov i Alisin zajednički tajni ključ.

Enkripcija


Public Key
Different Key
Secret Key
Aa
BbCc
DdEe
FfGg
Encryption
D$^*B
Ff/-*g
t#2@
1Zx3%
Decryption
Aa
BbCc
DdEe
FfGg
Plain Text
Cipher Text
Plain Text

Uparivanje na eliptičkoj krivoj (elliptic curve pairing)

Definition: Let E be an elliptic curve over a finite field K. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be additively-written subgroups of order p, where p is a prime number, of elliptic curve E, and let $g_1 \in \mathbb{G}_1, g_2 \in \mathbb{G}_2$ are the generators of groups $\mathbb{G}_1$ and $\mathbb{G}_2$ respectively. A map $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$, where $\mathbb{G}_T$ is a multiplicatively-written subgroup of K of order p, is called a elliptic curve pairing if satisfies the following conditions:

1. $e(g_1, g_2) \neq 1$,
2. $\forall R, S \in \mathbb{G}_1, \forall T \in \mathbb{G}_2 : e(R + S, T) = e(R, T) * e(S, T)$,
3. $\forall R \in \mathbb{G}_1, \forall S, T \in \mathbb{G}_2 : e(R, S + T) = e(R, S) * e(R, T)$.

The following properties of the elliptic curve pairing can be easily verified:

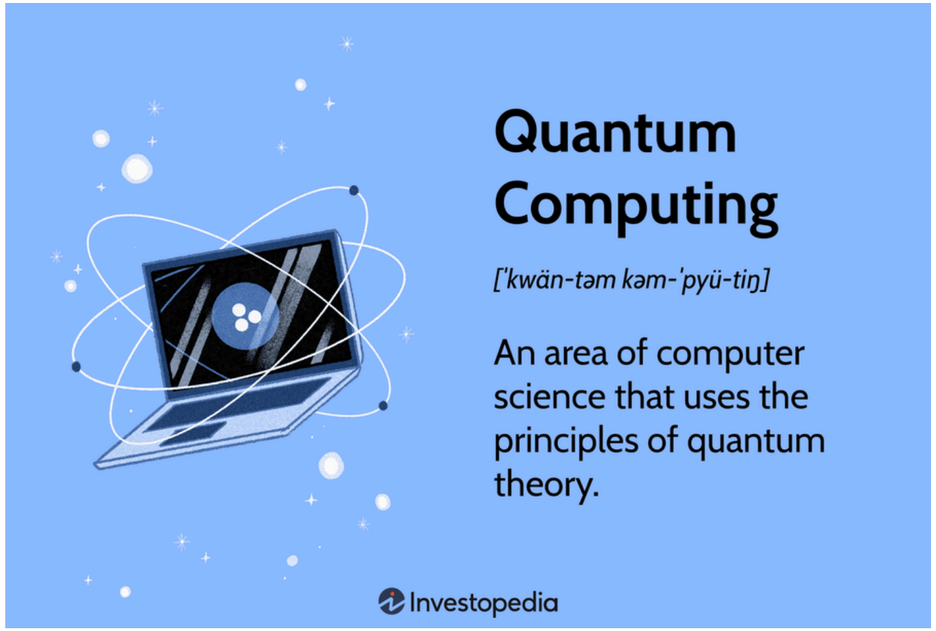
1. $\forall S \in \mathbb{G}_1, \forall T \in \mathbb{G}_2 : e(S, -T) = e(-S, T) = e(S, T)^{-1}$,
2. $\forall S \in \mathbb{G}_1, \forall T \in \mathbb{G}_2 : e(a * S, b * T) = e(b * S, a * T) = e(S, T)^{a*b}$.

Tipovi uparivanja

- **Type 1:** $\mathbb{G}_1 = \mathbb{G}_2$, and we say e is a symmetric bilinear map;
- **Type 2:** $\mathbb{G}_1 \neq \mathbb{G}_2$ and there exists an efficient homomorphism $\phi : \mathbb{G}_2 \to \mathbb{G}_1$, but no efficient one exists in the other direction;
- **Type 3:** $\mathbb{G}_1 \neq \mathbb{G}_2$ and there exists no efficient homomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$.

Post-kvantna kriptografija

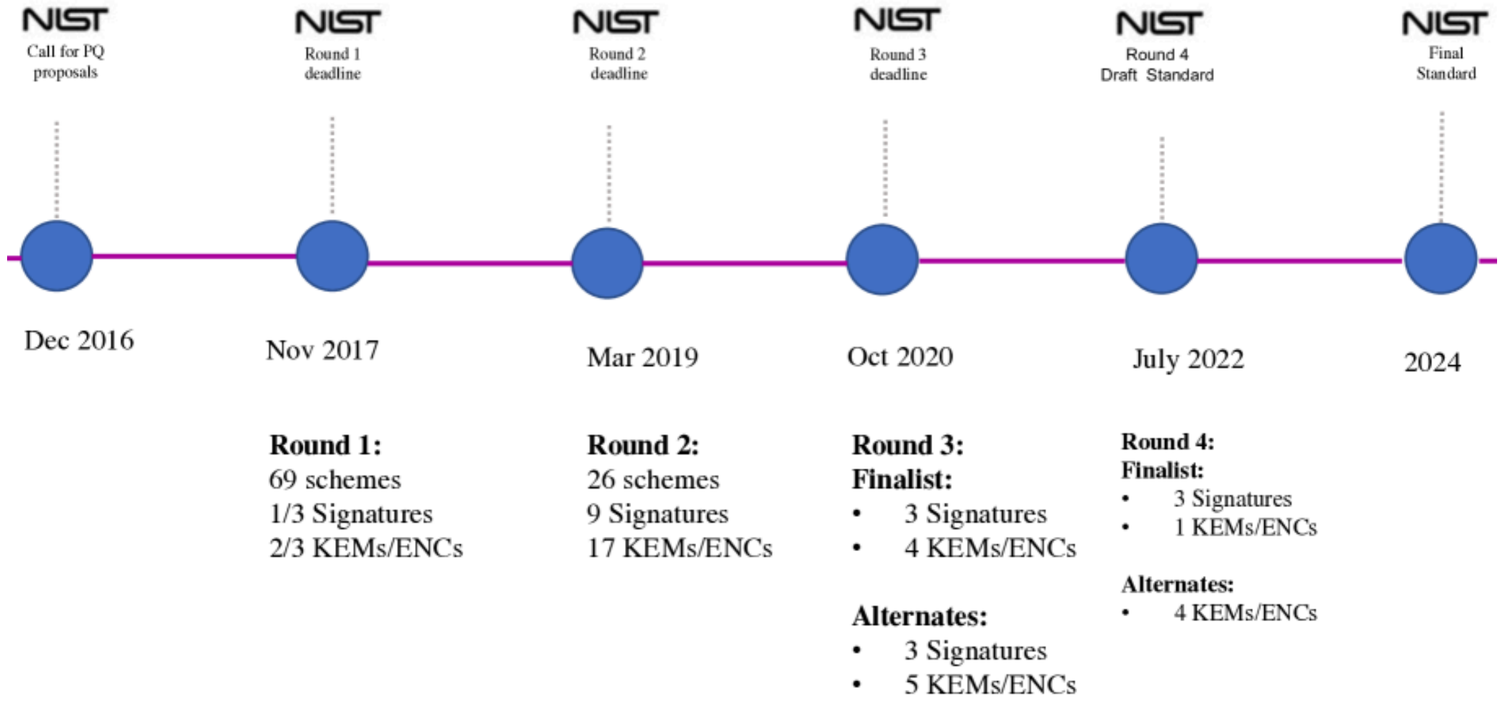
- Današnja asimetrična kriptografija se oslanja na sledeće probleme:
  - RSA: faktorizacija velikih brojeva
  - Diffie-Hellman: diskretni logaritam u Zp*
  - ECC: diskretni logaritam na eliptičkim krivama
- Dovoljno snažan kvantni računar mogao bi efikasno da reši ove probleme pomoću Šorovog (Peter Shor) algoritma.
- Zato se uvodi post-kvantna kriptografija: kriptografija koja se izvršava na običnim računarima, ali je dizajnirana da bude otporna i na kvantne računare.



Simetrična kriptografija nije razbijena na isti način.
AES i hash funkcije ostaju upotrebljivi, ali se obično preporučuju veći sigurnosni parametri.

Post-kvantna kriptografija

Godine 2017. Američki nacionalni institut za standarde i tehnologiju (NIST) pokrenuo je proces standardizacije novih kriptografskih algoritama koji bi bili otporni na napade kvantnih računara.



Post-kvantna kriptografija

Nakon nekoliko krugova evaluacije, NIST je izabrao sledeće post-kvantne algoritme:

Key Encapsulation Mechanisms (KEM):

- **CRYSTALS-Kyber (ML-KEM)**
  Module-Lattice-based Key Encapsulation Mechanism
  → Primarna post-kvantna zamena za mehanizme razmene ključa zasnovane na RSA, Diffie-Hellmanu i eliptičkim krivama.
- **HQC (Hamming Quasi-Cyclic)**
  Code-based Key Encapsulation Mechanism
  → Rezervni KEM zasnovan na kodovima, izabran kao dodatna alternativa za standardizaciju.

Algoritmi za digitalno potpisivanje:

- **CRYSTALS-Dilithium (ML-DSA)**
  Module-Lattice-based Digital Signature Algorithm
- **FALCON**
  Lattice-based Digital Signature Algorithm (NTRU lattice)
- **SPHINCS+ (SLH-DSA)**
  Stateless Hash-based Digital Signature Algorithm

Shortest Vector Problem (SVP) i Closest Vector Problem (CVP)

Lattice, odnosno rešetka, je skup tačaka koje dobijamo celobrojnim kombinovanjem baznih vektora. Mnogi post-kvantni algoritmi koriste činjenicu da su određeni problemi na rešetkama, veoma teški za rešavanje.
Dva najosnovnija problema su:

SVP (Shortest Vector Problem): među svim tačkama rešetke koje nisu 0, treba pronaći onu koja je najbliža koordinatnom početku.

CVP (Closest Vector Problem): data je rešetka i jedna tačka koja ne mora da pripada rešetki.
Cilj je pronaći tačku rešetke koja je najbliža toj zadatoj tački.

Zašto su ovi problemi teški?

SVP i CVP su laki za razumevanje, ali veoma teški za rešavanje u velikom broju dimenzija.

U kriptografiji se, koriste lattice-i sa stotinama ili hiljadama dimenzija. Broj mogućih kandidata tada raste eksponencijalnom brzinom.

Pored toga, baza lattice-a može biti veoma iskrivljena, pa najkraći vektor ili najbliža tačka nisu očigledni.

Zato su lattice problemi dobra osnova za post-kvantnu kriptografiju.

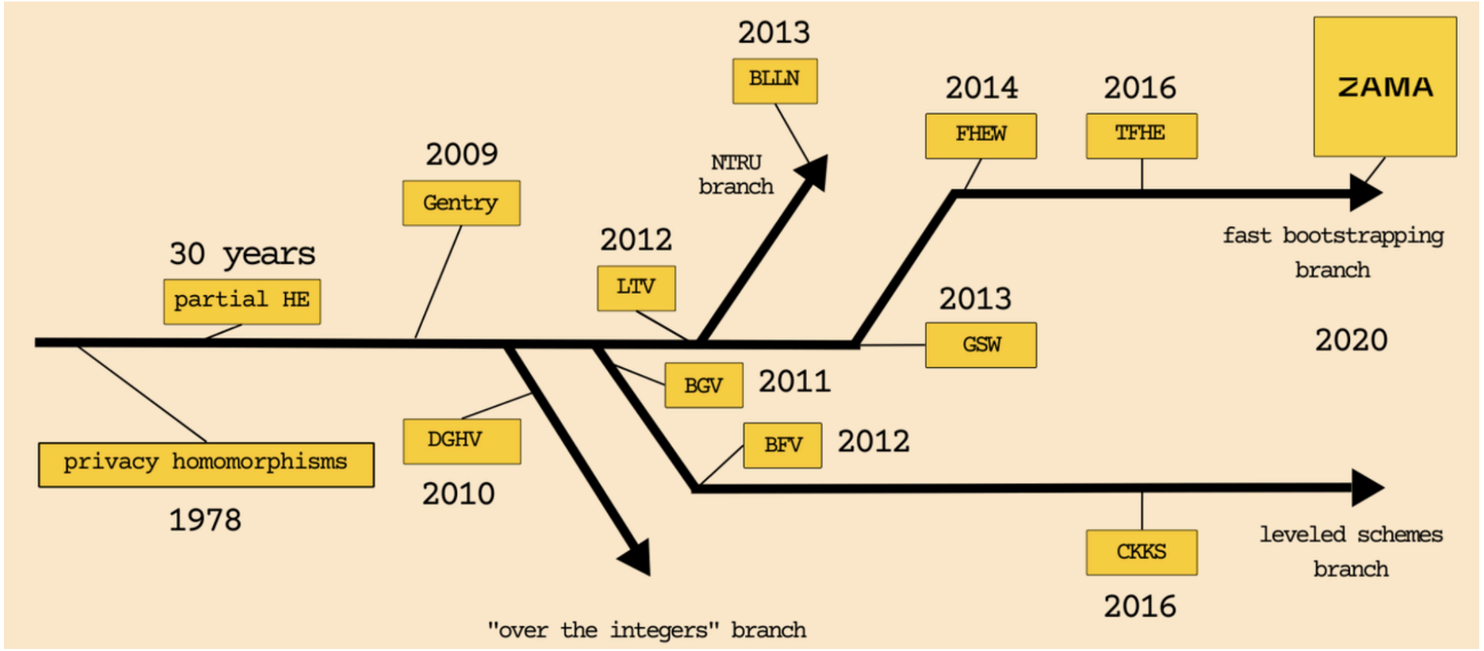
Lattice-based problemi u praksi

U praksi se ne koriste uvek direktno baš SVP i CVP, već njihove efikasnije i praktičnije varijante.
Najpoznatije porodice problema su:
LWE (Learning With Errors);
Ring-LWE;
Module-LWE;
SIS (Short Integer Solution);
Module-SIS;
NTRU problemi.

Ove varijante uvode dodatnu strukturu i optimizacije, kako bi algoritmi bili brži i pogodniji za implementaciju.

FHE (Fully Homomorphic Encryption)

Prvu potpuno homomorfnu šemu šifrovanja predstavio je Craig Gentry 2009. godine u svojoj doktorskoj disertaciji.
Od tada je razvijen veliki broj FHE šema, uz značajan napredak u efikasnosti, bootstrapping tehnikama i praktičnoj upotrebljivosti.
Važno je da se moderne FHE šeme zasnivaju na lattice-based problemima, odnosno na Module-LWE.
Danas razvoj FHE-a guraju i akademska zajednica i industrija, sa sve većim fokusom na praktične primene nad šifrovanim podacima.



ZKP (Zero Knowledge Proof)

ZKP sistemi kombinuju tri glavne komponente:

1. aritmetizaciju problema
2. commitment šemu
3. protokol za proveru tvrdnje nad komitovanim podacima

U zavisnosti od toga koje se komponente koriste, dobijamo različite porodice ZK sistema.

Dve najpoznatije porodice su:
SNARK (Succinct Non-interactive Argument of Knowledge)
STARK (Scalable Transparent Argument of Knowledge)

SNARK vs STARK

SNARK (Succinct Non-interactive Argument of Knowledge)

kratki dokazi
brza verifikacija
zahteva trusted setup (postoje i novije varijante bez trusted setup-a)
u klasičnim varijantama koristi eliptičke krive i pairings (postoje i post-kvantne varijante bez eliptičkih krivih)

Primeri:
Groth16
PLONK sa KZG commitmentima

STARK (Scalable Transparent Argument of Knowledge)

veći dokazi
transparentan setup
oslanja se na hash funkcije
otporan je na kvantne računare