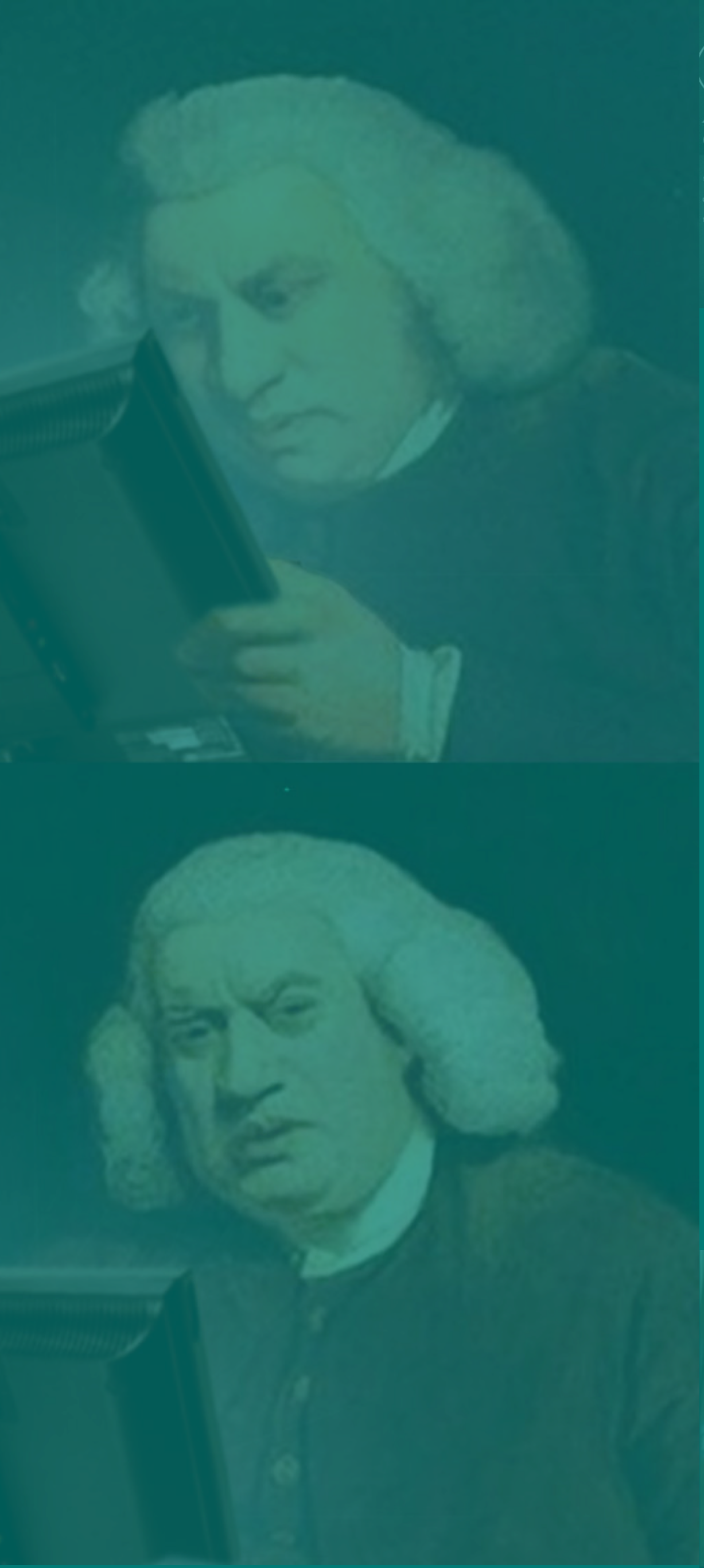
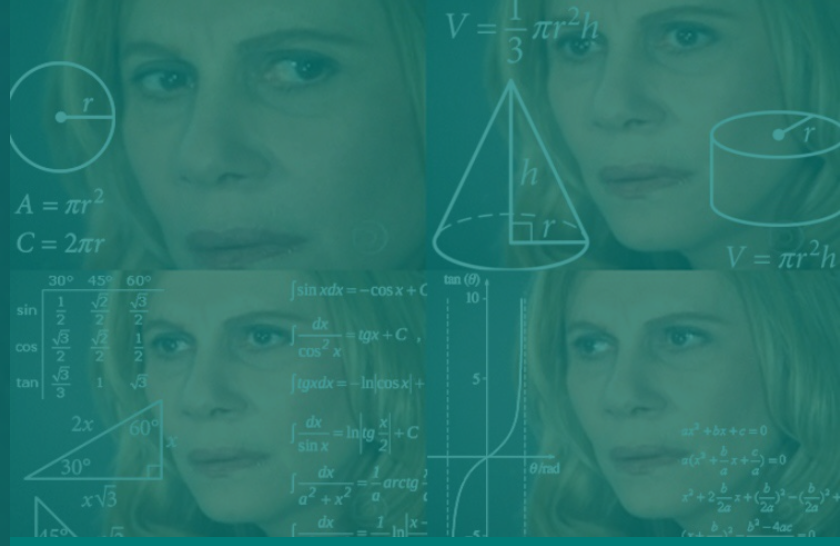
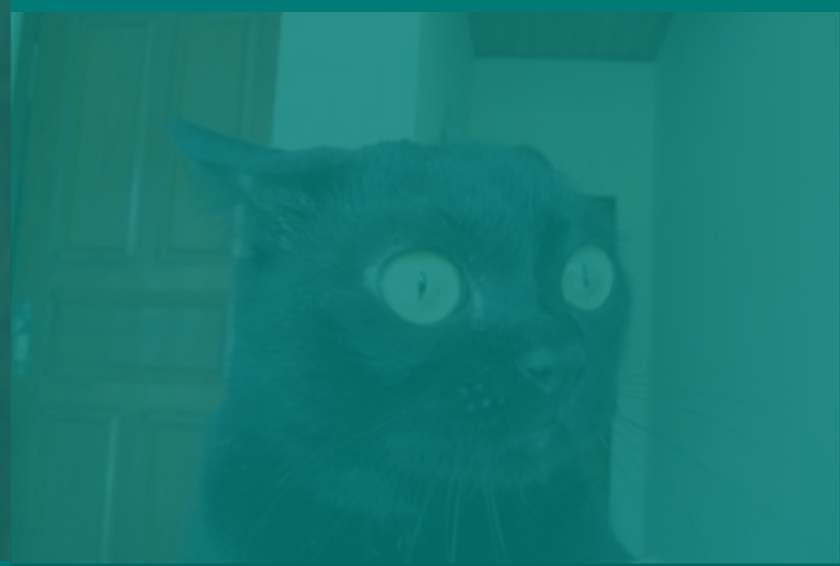




ALEKSANDAR VELJKOVIĆ

Kriptografija

BELEŠKE SA VEŽBI



ASIMETRIČNA KRIPTOGRAFIJA

Za razliku od simetrične kriptografije, koja podrazumeva postojanje jednog kjuca, asimetrična kriptografija uvodi postojanje još jednog ključa. Jedan ključ ostaje skriven od ostalih učesnika i naziva se *privatni ključ,* dok je drugi ključ poznat svim učesnicima i naziva se *javni ključ*. Ovakva postavka se može koristiti na dva načina:

• *Privatni ključ* se koristi za enkripciju, javni ključ za dekripciju

• *Javni ključ* se koristi za enkripciju, privatni ključ za dekripciju

U zavisnosti od cilja, odabira se jedan od dva navedena pristupa. Tri osnovne primene asimetrične kriptografije su:

• Enkripcija

• Razmena ključa

• Digitalni potpis

Upotreba asimetrične kriptografije široko je zastupljena, od HTTPS/SSL protokola koji se koriste na većini veb sajtova, do kriptovaluta gde javni ključevi predstavljaju identitete elektronskih novčanika. Isti algoritmi se mogu koristiti za više namena.

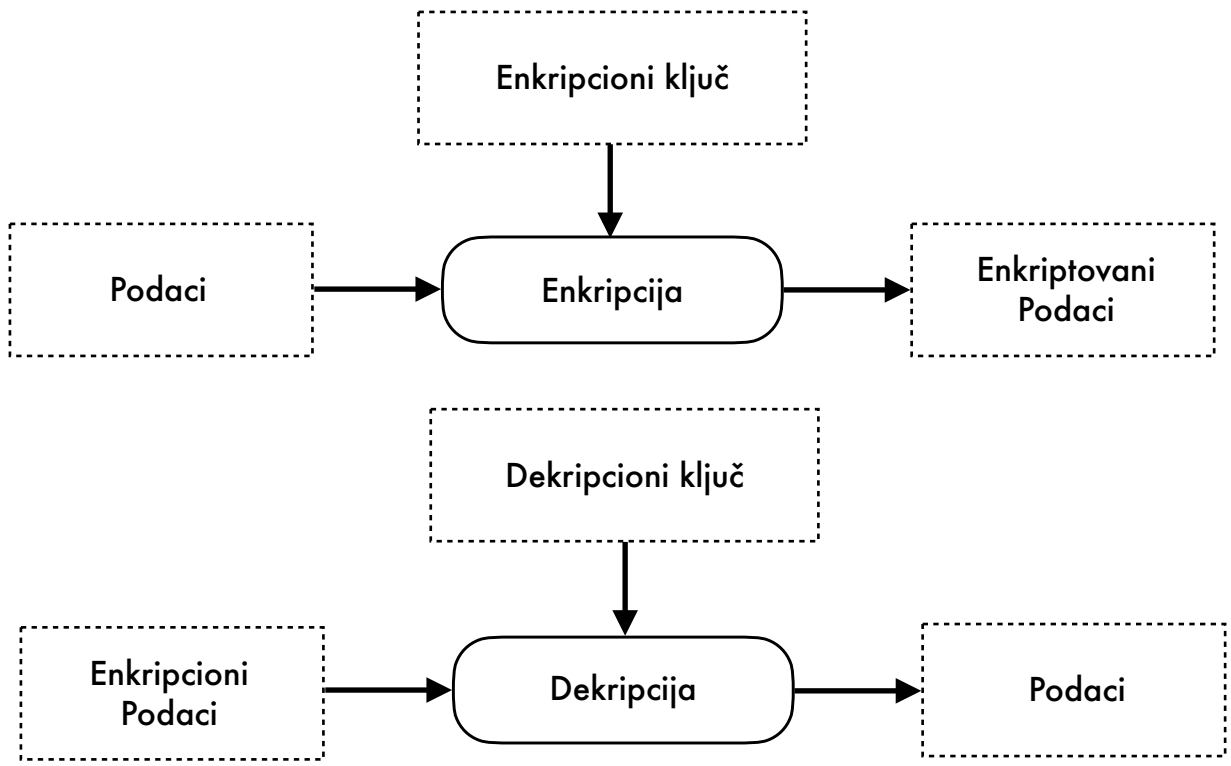
Jednom privatnom ključu odgovara jedan javni ključ i obrnuto. Poznavanjem javnog ključa izuzetno je teško rekonstruisati privatni ključ.

ENKRIPCIJA

Javni ključ - enkripcija, privatni ključ - dekripcija

Kao što je bio slučaj i kod simetričnih šifara, enkripcija podataka podrazumeva skrivanje podataka od svih učesnika koji nemaju ključ kojim se podaci mogu otkriti. Asimetrična kriptografija koristi *javni ključ* primaoca za *enkriptovanje* podataka a *privatni ključ* primaoca za *dekriptovanje* podataka. Iako je moguće, asimetrična enkripcija nije pogodna za enkriptovanje

velike količine podataka zbog svoje brzine, za razliku od simetričnih šifara koje su i do 1000 puta brže od asimetričnih. Iz tog razloga, simetrična kriptografija se koristi u specifičnim okolnostima gde je količina podataka mala i gde osobine asimetrične kriptografije daju najveći doprinos. Čest scenario je enkripcija podataka simetričnim ključem a zatim korišćenje algoritama asimetrične kriptografije za enkripciju simetričnog **ključa** koji se zajedno sa enkriptovanim podacima šalje primaocu.

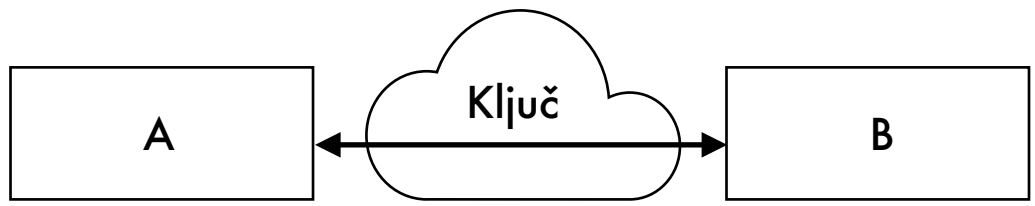


Dijagram asimetrične enkripcije/dekripcije

RAZMENA KLJUČA

privatni ključ - enkripcija/dekripcija, javni ključ - enkripcija

Razmena ključa može se smatrati posebnim oblikom enkripcije, gde je cilj skriveno (nerazumljivo za ostale učesnike) dogovoriti simetrični ključ koji će se nadalje koristiti za enkripciju podataka koji se prenose između učesnika koji učestvuju u razmeni. U zavisnosti od algoritma, ključ se može preneti enkripcijom privatnim ključem od strane oba učesnika, bez potrebe dekripcije (npr. *Diffie-Hellman* razmena) dok se kod drugih algoritama vrši enkripcija javnim ključem primaoca i dekripcija privatnim ključem primaoca, kao što je uobičajen slučaj kod enkripcije (npr. *RSA*).

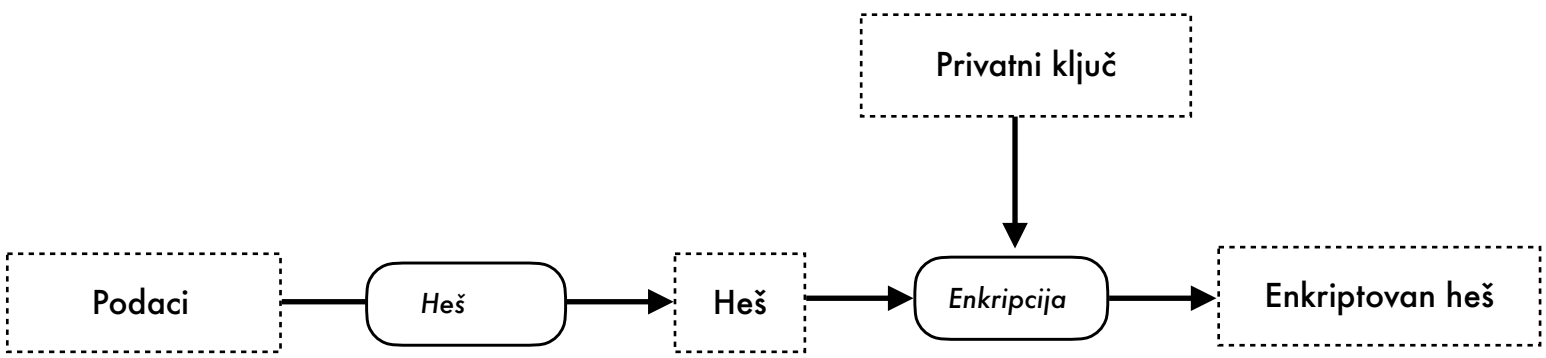


Dijagram razmene ključa

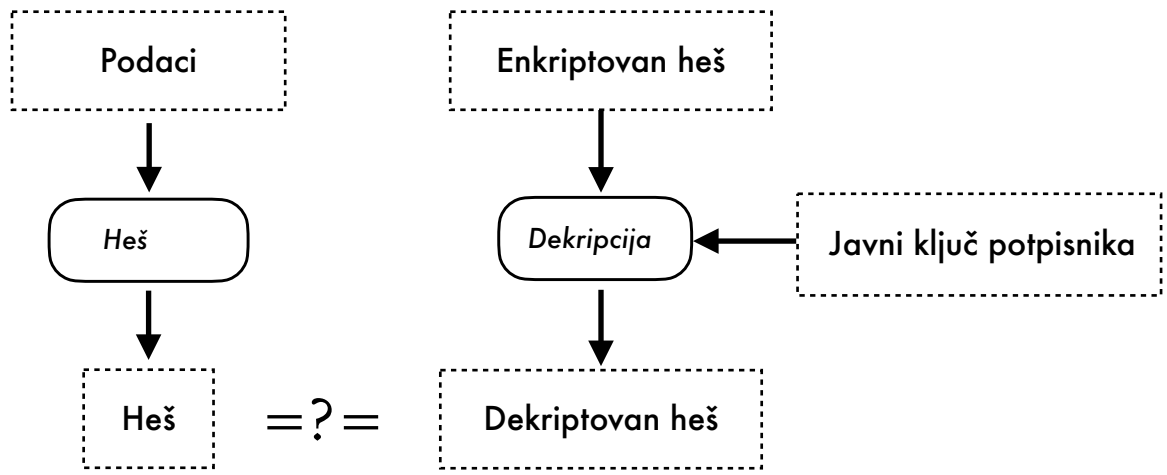
DIGITALNI POTPIS

privatni ključ - enkripcija, javni ključ - dekripcija

Digitalni potpis se koristi za potrebe identifikacije. Učesnik svojim privatnim ključem potpisuje (enkriptuje) podatke (ugovore, izjave, potvrde, …) dok je javni ključ učesnika registrovan kod autorizacionog tela i koristi se za proveru (dekripciju) podataka radi utvrđivanja da je podatke zaista enkriptovao vlasnik odgovarajućeg privatnog ključa. Kako je navedeno da se asimetrična kriptografija ne koristi za enkripciju velikih podataka, praksa je da se podaci prvobitno heširaju nekom od heš funkcija koja podatke preslikava u mali domen fiksne dužine a zatim se dobijeni rezultat heširanja enkriptuje privatnim ključem (npr. *SHA256 + RSA*), pri čemu rezultujući enkriptovani heš predstavlja digitalni potpis podataka.



Protokol generisanja digitalnog potpisa



Protokol provere digitalnog potpisa

RSA

RSA (*Rivest, Shamir, Adleman*) algoritam, dobio je naziv prema trojici naučnika koji su ga konstruisali. U praksi se najčešće koristi za enkripciju i generisanje digitalnog potpisa, ali je moguće i korišćenje RSA za potrebu razmene ključa. Bezbednost algoritma zasniva se na težini *faktorizacije velikih brojeva*.

Postavka (tajna)

1. Generisati dva velika prosta broja:
   p, q
2. Izračunati proizvod brojeva p i q:
   $n = p \cdot q$
3. Izračunati vrednost Ojlerove funkcije:
   $\phi(n) = \phi(p \cdot q) = \phi(p) \cdot \phi(q) = (p - 1) \cdot (q - 1)$

   Napomena: Lakoća izračunavanja dolazi iz osobine da su p i q poznati prosti brojevi, u suprotnom bi vreme računanja zavisilo od vremena potrebnog za faktorizaciju broja n.

4. Odabrati proizvoljan broj e sa osobinom da je uzajamno prost sa $\phi(n)$:
   $e \rightarrow NZD(e, \phi(n)) = 1$

   Napomena: Odabrani broj e predstavljaće enkripcioni ključ.

5. Izračunati broj d kao multiplikativni inverz broja e po modulu $\phi(n)$:
   $d \rightarrow d \cdot e \equiv 1(mod\ \phi(n)) \rightarrow d = e^{-1}(mod\ \phi(n))$

Enkripcija / dekripcija

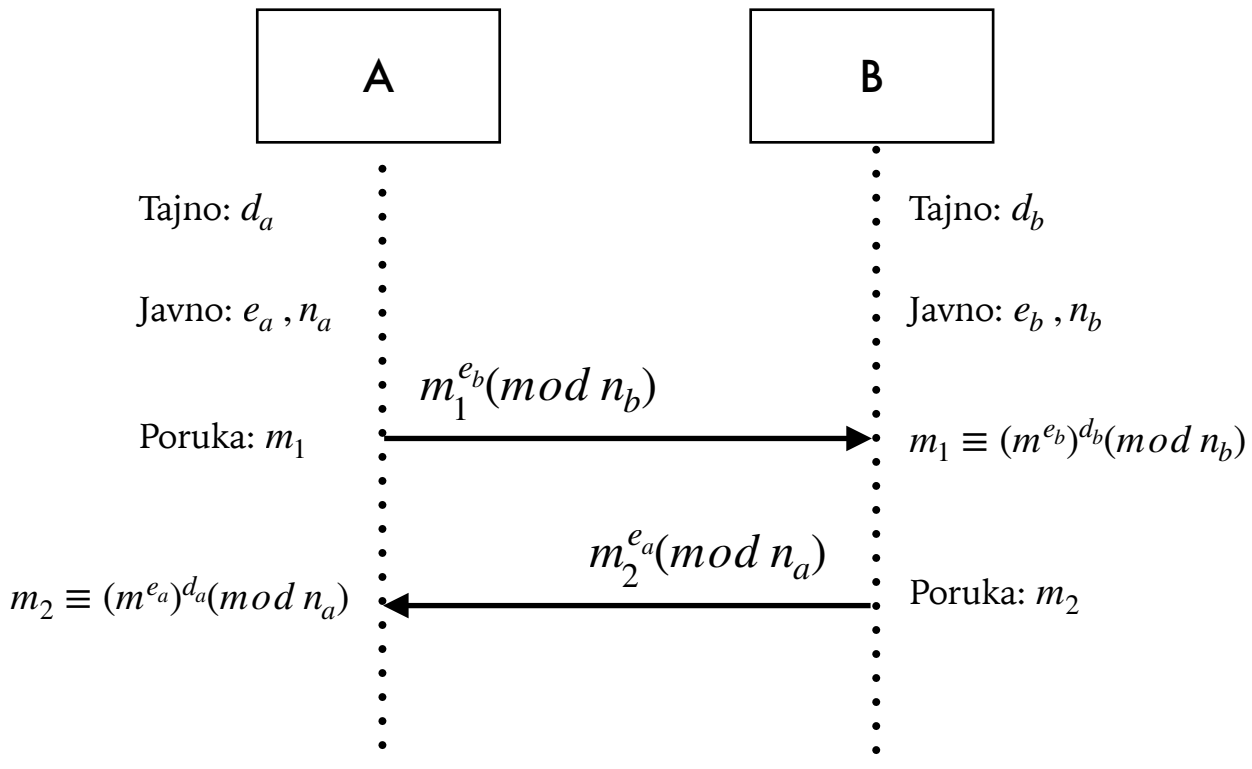
Enkripcija poruke m: $m_e = m^e(mod\ n)$

Dekripcija enkriptovane poruke:
$m_e^d(mod\ n) \equiv (m^e)^d(mod\ n) \equiv m^{ed}(mod\ n) \equiv m^{e^{-1}e}(mod\ n) \equiv m^1(mod\ n) \equiv m(mod\ n)$

Javni podaci: (n, e)

Tajni podaci: d

RSA Protokol



Dijagram protokola enkripcije podataka RSA algoritmom

DIFFIE-HELLMAN

Diffie-Hellman algoritam za razmenu ključa koristi se za razmenu simetričnog ključa između učesnika. Bezbednost algoritma leži u težini izračunavanja *diskretnog logaritma*.

Postavka (tajna)

0. **Javni parametri:** *g, n,* gde je *g* generator mukltiplikativne grupe prostog reda *n* (***n* je prost broj**)

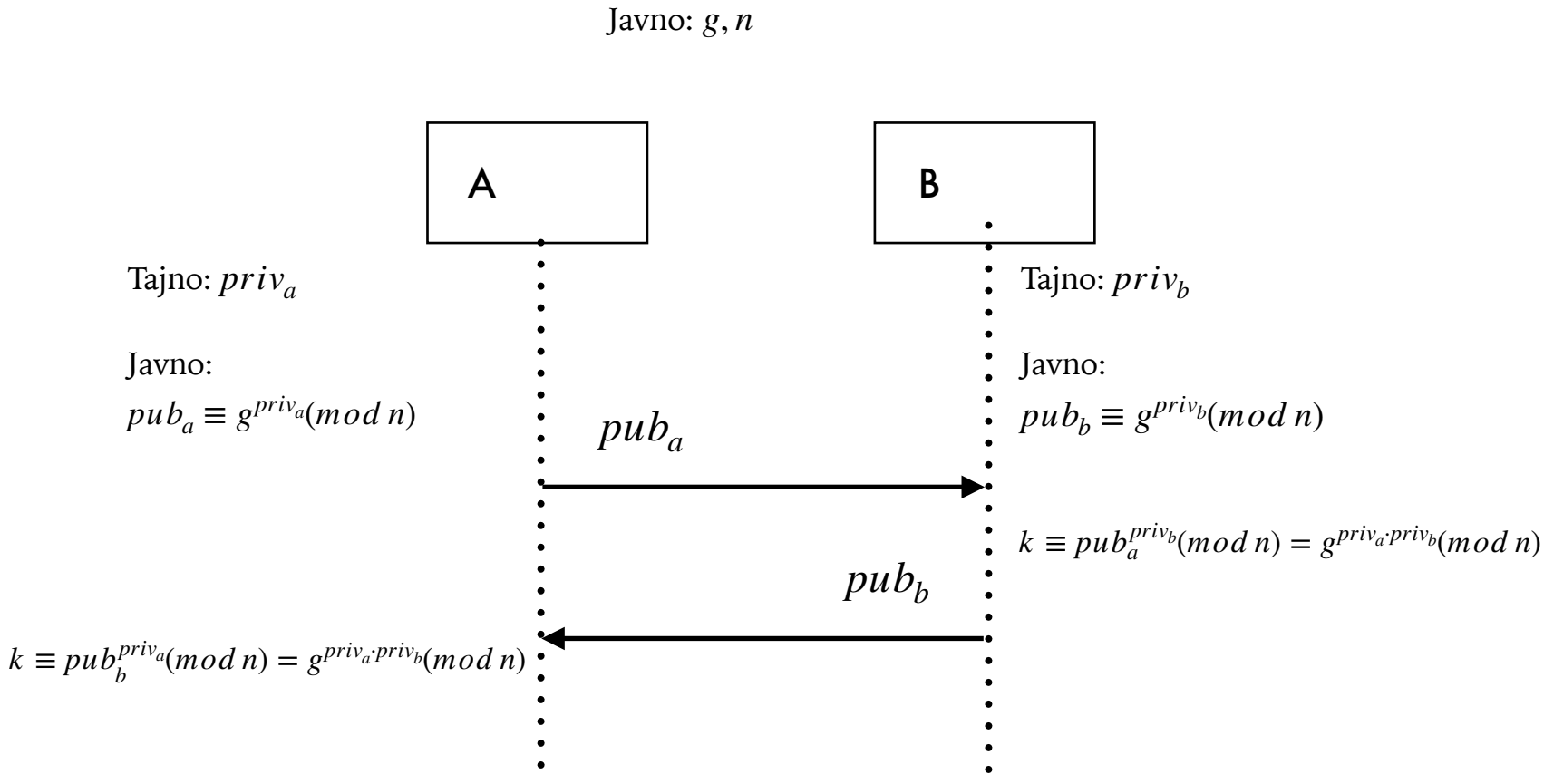
1. Učesnik A generiše privatni ključ pr_a kao slučajni broj iz intervala $(1,n)$
2. Učesnik B generiše privatni ključ pr_b kao slučajni broj iz intervala $(1,n)$
3. Učesnik A generiše javni ključ $pub_a = g^{pr_a} (mod\ n)$
4. Učesnik B generiše javni ključ $pub_b = g^{pr_b} (mod\ n)$

Razmena ključa

1. Učesnik A šalje svoj javni ključ učesniku B
2. Učesnik B računa vrednost $k \equiv pub_a^{pr_b} (mod\ n) \equiv (g^{pr_a})^{pr_b} (mod\ n) \equiv g^{pr_a \cdot pr_b} (mod\ n)$
3. Učesnik A šalje svoj javni ključ učesniku B
4. Učesnik B računa vrednost $k \equiv pub_a^{pr_b} (mod\ n) \equiv (g^{pr_a})^{pr_b} (mod\ n) \equiv g^{pr_a \cdot pr_b} (mod\ n)$

Dobijena vrednost *k* predstavlja zajednički ključ za simetričnu enkripciju podataka koji će se razmenjivati između učesnika.

Diffie-Hellman Protokol



Dijagram protokola razmene ključa Diffie-Hellman algoritmom

ELGAMAL

ElGamal algoritam za enkripciju podataka. Nastao je kao proširenje *Diffie-Hellman* algoritama i zbog svoje osnove, bezbednost algoritma takođe leži u težini izračunavanja *diskretnog logaritma*. Algoritam se uglavnom koristi za potrebe enkripcije.

Postavka (tajna)

0. **Javni parametri:** *g, n,* gde je *g* generator mukltiplikativne grupe prostog reda *n* (***n* je prost broj**)

1. Učesnik A generiše privatni ključ pr_a kao slučajni broj iz intervala $(1,n)$

2. Učesnik B generiše privatni ključ pr_b kao slučajni broj iz intervala $(1,n)$

3. Učesnik A generiše javni ključ $pub_a = g^{pr_a}(mod\ n)$

4. Učesnik B generiše javni ključ $pub_b = g^{pr_b}(mod\ n)$

Enkripcija / dekripcija

1. Učesnik A generiše slučajni broj k iz intervala $(1,n)$
   $k = random(1,n)$

2. Učesnik A izračunava vrednost maskirajućeg ključa
   $mask_k \equiv g^k(mod\ n)$

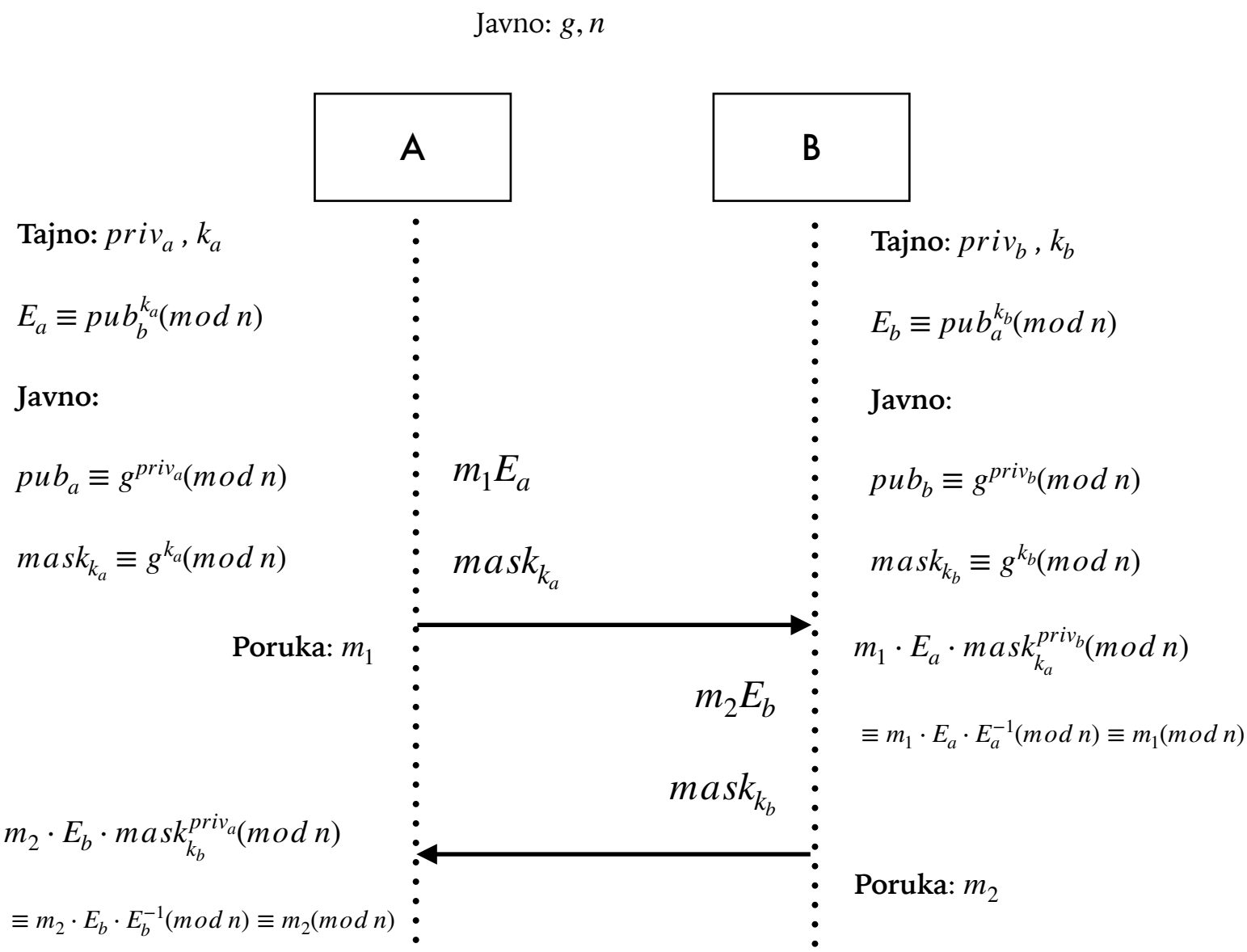
3. Učesnik A poruku m enkriptuje pomoću maskirajućeg i javnog ključa učesnika B
   $m_e \equiv m \cdot pub_b^k(mod\ n) \equiv m \cdot (g^{priv_b})^k(mod\ n) \equiv m \cdot g^{k \cdot priv_b}(mod\ n) \equiv mE(mod\ n)$

4. Učesnik A šalje enkriptovanu poruku m_e učesniku B, zajedno sa maskirajućim ključem
   $A - (m_e,\ mask_k) \rightarrow B$

5. Učesnik B izračunava vrednost E, pomoću maskirajućeg ključa i svog privatnog ključa
   $E \equiv mask_k^{priv_b}(mod\ n) \equiv (g^k)^{priv_b}(mod\ n) \equiv g^{priv_b \cdot k}(mod\ n)$

6. Učesnik B izračunava multiplikativni inverz vrednosti E po modulu n i njime množi maskiranu poruku dobijenu od učesnika A
   $m_e \cdot E^{-1}(mod\ n) \equiv m \cdot E \cdot E^{-1}(mod\ n) \equiv m(mod\ n)$

ElGamal Protokol



Dijagram protokola enkripcije ElGamal algoritmom

MESI-OMURA

Iako algoritam nije predstavnik ni grupe simetričnih niti asimetričnih algoritama, zbog sličnih algebarskih osnova i snage koja se zasniva na težini nalaženja diskretnog logaritma, algoritam Mesi-Omura biće naveden u ovom poglavlju. Zbog svog karakterističnog protokola koji se odvija u tri koraka razmene poruka, ovaj algoritam pripada grupi *Three-pass* protokola.

Postavka:

0. **Javni parametri**: Polje F_q, q veliki broj.

1. Učesnik A bira slučajan broj e_A iz polja F_q , uzajamno prost sa $q-1$

2. Učesnik A računa vrednost broja d_A, za koji važi:
   $e_A \cdot d_A \equiv 1(mod\ q-1) \rightarrow d_A \equiv e_A^{-1}(mod\ q-1)$

3. Učesnik B takođe bira slučajan broj e_B iz polja F_q , uzajamno prost sa $q-1$

4. Učesnik B računa vrednost broja d_B, za koji važi:
   $e_B \cdot d_B \equiv 1(mod\ q) \rightarrow d_B \equiv e_B^{-1}(mod\ q)$

Razmena ključa

1. Učesnik A vrednost ključa k enkriptuje pomoću broja e_A:
   $k \rightarrow k^{e_A}(mod\ q)$

2. Učesnik A šalje enkriptovan ključ k učesniku B

3. Učesnik B enkriptuje dobijenu vrednost svojim ključem e_B:
   $k^{e_A}(mod\ q) \rightarrow (k^{e_A})^{e_B}(mod\ q) \equiv k^{e_A e_B}(mod\ q)$

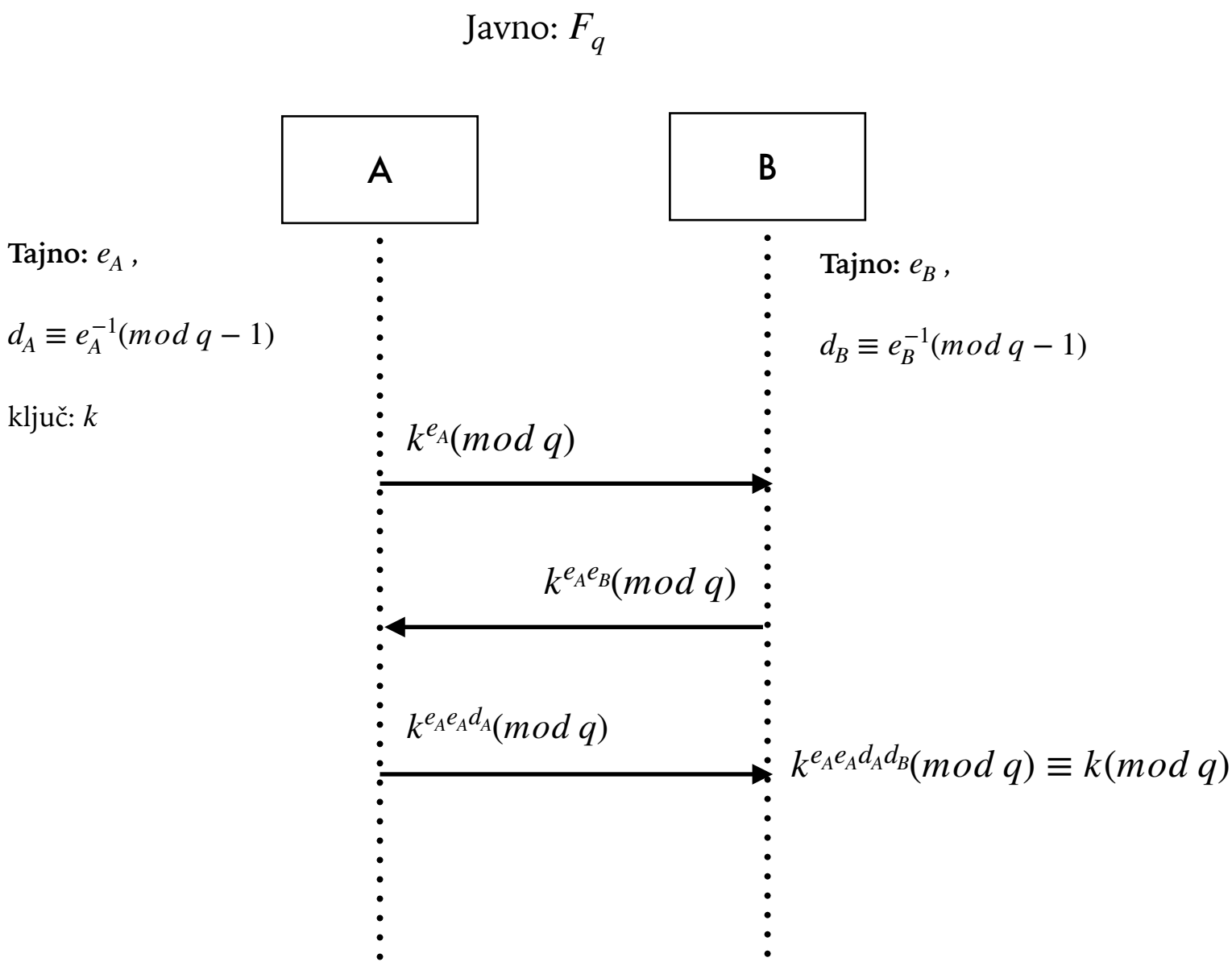
4. Učesnik B rezultat svoje enkripcije vraća učesniku A

5. Učesnik A vrši dekripciju dobijene vrednost pomoću d_A:
   $k^{e_A e_B}(mod\ q) \rightarrow (k^{e_A e_B})^{d_A}(mod\ q) \equiv k^{e_A e_B d_A}(mod\ q) \equiv k^{e_A e_B e_A^{-1}}(mod\ q) \equiv k^{e_B}(mod\ q)$

6. Učesnik A šalje izračunatu vrednost učesniku A

7. Učesnik B vrši dekripciju dobijene vrednosti pomoću d_B i otkriva vrednost ključa:
   $k^{e_B}(mod\ q) \rightarrow (k^{e_B})^{d_B}(mod\ q) \equiv k^{e_B d_B}(mod\ q) \equiv k^{e_B e_B^{-1}}(mod\ q) \equiv k^1(mod\ q) \equiv k(mod\ q)$

Mesi-Omura Protokol



Dijagram protokola enkripcije ElGamal algoritmom

KRIPTOGRAFIJA NAD ELIPTIČKIM KRIVAMA

Eliptičke krive su kubne krive oblika $y^2 - a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$. Za potrebe kriptografije, upotrebi su i krive sa nešto kraćim zapisom, čiji su koeficijenti $a_1 = a_2 = a_3 = 0$ pa je kriva oblika $y^2 = x^3 + a_4x + a_6$ ili u uobičajenom zapisu $y^2 = x^3 + ax + b$ (Vajerštrasova kriva). Iako su ove krive poznate u matematici kao beskonačni skupovi realnih tačaka, u kriptografiji se definišu nad konačnim poljima (vrednosti x i y su iz konačnih skupova, u oznaci $y^2 = x^3 + ax + b\ (mod\ p)$). Iz tog razloga, neophodno je definisati polje tačaka i koeficijenata.

Razlog korišćenja eliptičkih krivih, pored postojećih kriptosistema sa javnim ključem koji se zasnivaju na teškoći rešavanja problema diskretnog logaritma, je taj što se sa istom dužinom ključa u sistemu sa eliptičkom krivom obezbeđuje znatno veća bezbednost nego što je to slučaj sa prethodnim kriptosistemima. Primer takve paralele je da je za ostvarivanje bezbednosti nivoa RSA sa 2048-bitnim ključem potreban samo 256-bitni ključ pri enkripciji zasnovanoj na eliptičkoj krivoj. Pri konstrukciji Vajerštrasove krive (prema standardu FIPS 184-6) za koeficijent a uzima se vrednost $a \equiv -3(mod\ p)$, gde je p red polja nad kojim je generisana kriva pa su te jednačine oblika $y^3 = x^2 - 3x + b\ (mod\ p)$

Neke standardne eliptičke krive su definisane od strane NIST (*National Institute of Science and Technology*) od kojih su neke dodatno preporučene od strane NSA (*National Security Agency*). To su krive NIST P-256 i NIST P-384 sa koeficijentima:

NIST P-256:

$p = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$

$a \equiv -3(mod\ p)$

$b = 41058363725152142129326129780047268409114441015993725554835256314039467401291$

NIST P-384:

$$p = 2^{384} - 2^{128} - 2^{96} + 2^{32} - 1$$

$$a \equiv -3 (mod\, p)$$

b = 27580193559959705877849011840389048093056905856361568521428707301988689241309860865136260764883745107765439761230575

Dodatne korisne preporuke za odabir koeficijenata krive mogu se pronaći na adresi: www.secg.org/SEC2-Ver-1.0.pdf

Pored Vajerštrasovih krivih, u upotrebi su i *Koblitz* krive (koje nose ima po Nilu Koblicu, jednom od autora ove familije krivih). Ovaj tip krivih definiše se širom jednačinom $y^2 - xy = x^3 + ax^2 + b\ (mod\, p)$ kod koje su samo koeficijenti $a_2 = a_3 = 0$ dok je $a_1 = 1$. Osobina Koblic krivih je da tačke krive nisu simetrične u odnosu na x-osu (kao što je slučaj kod Vajerštrasovih), ali i da je vršenje operacija nad tačkama krive nešto efikasnije.

Bitcoin i Ethereum koriste Koblic krive za generisanje privatnih i javnih ključeva iz kojih se izvodi adrese novčanika.

U daljem tekstu će diskusija biti zasnovana na Vajerštrasovim krivama.

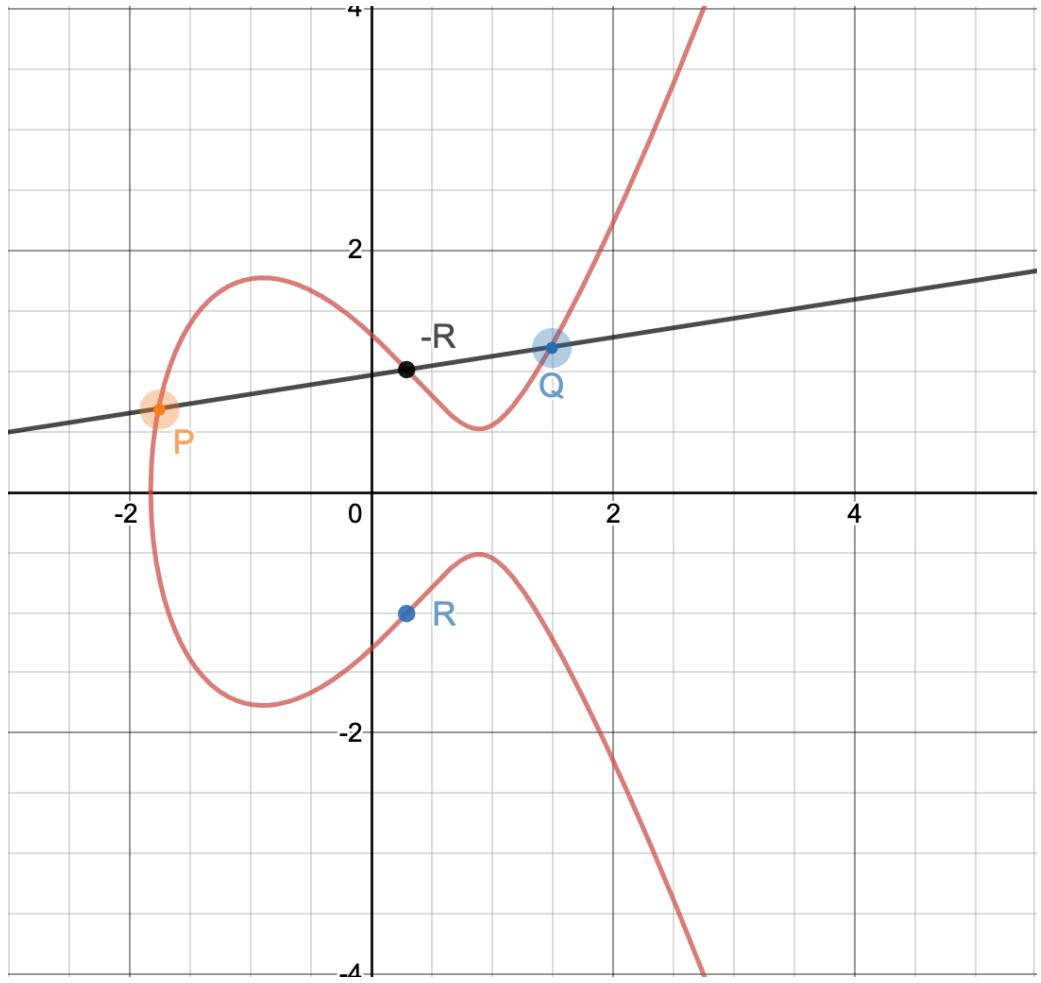
OPERACIJE NAD TAČKAMA ELIPTIČKE KRIVE

Nad tačkama eliptičke krive, definisane nad konačnom polju, osnovna je operacija sabiranja tačaka. Zbir tačka krive P i Q definiše se kao tačka simetrična (u odnosu na x-osu) presečnoj tački jednačine eliptičke krive i jednačine prave koja prolazi kroz tačke P i Q.

$$P + Q = R$$

Specijalni slučaj predstavljaju tangente krive, kod kojih su tačke P i Q jednake (tangenta prolazi kroz jednu istu tačku. Pri tom slučaju, operacija sabiranja naziva se operacijom dupliranja tačke.

$$P + P = 2P$$



Primer zbira tačaka P + Q, rezultujuće tačke preseka sa eliptičkom krivom -R i njene projekcije u odnosu na x-osu R.

Posebno, uvodi se nova tačka, takozvana beskonačna tačka, koja predstavlja tačku susreta svih horizontalnih pravih i krive u beskonačnosti i definiše se kao neutral za sabiranje. Beskonačna tačka biće u daljem tekstu označena simbolom *0*.

Pravila koja važe za sabiranja sa beskonačnom tačkom su:

$$P + 0 = P$$

$$0 + P = 0$$

$$P + (-P) = 0$$

$$0 + 0 = 0$$

Oduzimanje se svodi na sabiranje sa inverznom tačkom, pri čemu je inverzna tačka tačke $Q(x_q, y_q)$ tačka $-Q = (x_q, -y_q)$.

IZRAČUNAVANJE ZBIRA DVE RAZLIČITE TAČKE NA ELIPTIČKOJ KRIVOJ

Neka su $P = (x_p, y_p)$ i $Q = (x_q, y_q)$ **različite** tačke na eliptičkoj krivoj sa jednačinom $y^2 = x^3 + Ax + B$. Neka je tačka $R = (x_r, y_r)$ rezultat zbita tačaka P i Q o odnosu na navedenu eliptičku krivu.

Za pravu $y = kx + n$ koja prolazi kroz tačke P i Q važi:

- $k = \dfrac{y_q - y_p}{x_q - x_p}$
- $n = -kx_p + y_p$

pa je konačna jednačina tražene prave: $y = \dfrac{y_q - y_p}{x_q - x_p} + (-\dfrac{y_q - y_p}{x_q - x_p} \cdot x_p + y_p)$

Potrebno je naći tačke preseka prave i eliptičke krive. Poznato je da prava sigurno preseca krivu u tačkama P i Q (kroz koje prolazi i pomoću kojih je definisana). Ostaje da se izračuna treća tačka preseka. Ako su P i Q različite realne tačke, po definiciji kubne krive postoji treća realna presečna tačka $R'(x'_r, y'_r)$. Treća tačka preseka izračunava se kao rešenje sistema:

$$y = kx + n$$

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

Iz prve jednačine sledi da je:

$$y^2 = (kx + n)^2 = k^2x^2 + 2kxn + n^2$$

Kako je iz polazne jednačine krive $y^2 = x^3 + Ax + B$ i $y^2 = k^2x^2 + 2kxn + n^2$ sledi da i desne strane obe jednačine moraju biti jednake pa je:

$$x^3 + Ax + B = k^2x^2 + 2kxn + n^2$$

$$\text{ili } x^3 + Ax + B - k^2x^2 - 2kxn - n^2 = x^3 - k^2x^2 + (A - 2kn)x + B - n^2 = 0$$

Kako dobijena jednačina $x^3 - k^2x^2 + (A - 2kn)x + B - n^2 = 0$ predstavlja polinom sa nulama u presečnim tačkama prave i eliptičke krive, oa se može izraziti i kao proizvod korena kubne jednačine:

$$(x - x_p)(x - x_q)(x - x'_r) = 0$$

ili nakon množenja:

$$(x - x_p)(x - x_q)(x - x'_r)$$

$$= (x^2 - xx_q - xx_p + x_px_q)(x - x'_r)$$

$$= x^3 - x^2x'_r - x^2x_q + xx_qx'_r - x^2x_p + xx_px'_r + xx_px_q - x_px_qx'_r$$

$$= x^3 - (x_p + x_q + x'_r)x^2 + (x_px_q + x_px'_r + x_qx'_r)x - x_px_qx'_r = 0$$

Sada je potrebno izjednačiti koeficijente iz poslednje i prethodne jednačine.

Uz x^3 u obe jednačine stoji 1. Uz x^2 u prvoj jednačini je $-k^2$ dok je u drugoj $-(x_p + x_q + x'_r)$ te ovi koeficijenti moraju biti jednaki:

$$-k^2x^2 = -(x_p + x_q + x'_r)x^2$$

$$k^2 = x_p + x_q + x'_r$$

$$x'_r = k^2 - x_p - x_q$$

U kompletno razvijenom obliku, koordinata x_r jednaka je:

$$x_r = (\frac{y_q - y_p}{x_q - x_p})^2 - x_p - x_q$$

gde su (x_p, y_p) i (x_q, y_q) koordinate poznatih polaznih tački P i Q.

Ovim postupkom pronađena je prva koordinata presečne tačke $R'(x'_r, y'_r)$. Druga koordinata, y'_r lako se izračunava iz jednačine prave:

$$y'_r = k \cdot x'_r + n$$

ili u razvijenom obliku:

$$y'_r = \frac{y_q - y_p}{x_q - x_p} \cdot x'_r - (\frac{y_q - y_p}{x_q - x_p})x_p + y_p$$

gde je x'_r koordinata tačke $R'(x'_r, y'_r)$ izračunata u prethodnom koraku.

Kako je rezultat sabiranja dve tačke na eliptičkoj krivoj tačka koja je **simetrična** u odnosu na treću tačku preseka prave koja prolazi kroz tačke-sabirke i krive, tražena tačka R za koju važi $P + Q = R$ simetrična je u odnosu na presečnu tačku R' te dele istu x koordinatu dok je y koordinata inverzna. Konačno:

$$P + Q = R = (x'_r, -y'_r)$$

IZRAČUNAVANJE VREDNOSTI DUPLIRANE TAČKE

Kako je napomenuto, moguć je slučaj kada su obe tačke pri sabiranju jednake, te prava seče eliptičku krivu kroz dve tačke umesto kroz tri. Ova situacija predstavlja specijalni slučaj sabiranja gde je:

$$P(x_p, y_p) = Q(x_q, y_q) \rightarrow x_p = x_q,\ y_p = y_q$$

pa bi onda, prema prethodnom računu, x koordinata presečne tačka bila jednaka:

$$x'_r = (\frac{y_q - y_p}{x_q - x_p})^2 - x_p - x_q$$

$$= (\frac{y_p - y_p}{y_p - y_p})^2 - x_p - x_p$$

$$= (\frac{0}{0})^2 - 0$$

što je nedefinisano. Ipak, kako je poznato da je tražena prava zapravo tangenta eliptičke krive koja prolazi kroz tačku $P(x_p, y_p)$, koeficijent prave dobija se računanjem izvoda:

$$\frac{\delta y}{\delta x} \cdot / \ y^2 = (x^3 + Ax + B)$$

$$\frac{\delta y}{\delta x} 2y = 3x^2 + A$$

$$\frac{\delta y}{\delta x} = \frac{3x^2 + A}{2y}$$

Što je u tački $P(x_p, y_p)$:

$$k = \frac{\delta y}{\delta x} = \frac{3x_p^2 + A}{2y_p}$$

pa je za sad jednačina prave oblika:

$$y = kx + n$$

$$y = \frac{3x_p^2 + A}{2y_p} x + n$$

Parametar n može se dobiti ponovnim uvrštavanjem koordinata tačke $P(x_p, y_p)$:

$$y_p = \frac{3x_p^2 + A}{2y_p} x_p + n$$

$$n = y_p - \frac{3x_p^2 + A}{2y_p} x_p$$

Konačno, jednačina prave je:

$$y = \frac{3x_p^2 + A}{2y_p} x + y_p - \frac{3x_p^2 + A}{2y_p} x_p$$

$$y = \frac{3x_p^2 + A}{2y_p} (x - x_p) + y_p$$

Tačka preseka dobijene prave i eliptičke krive računa se kao u prethodnom slučaju:

$$x^3 - k^2x^2 + (A - 2kn)x + B - n^2 = 0$$

Osim što u ovom slučaju polinom preseka ima dvostruki koren, jer su dve nule polinoma u istoj tački:

$$(x - x_p)(x - x_p)(x - x'_r) = (x - x_p)^2(x - x'_r) = 0$$

Nakon množenja:

$$(x - x_p)^2(x - x'_r)$$

$$= (x^2 - 2xx_p + x_p^2)(x - x'_r)$$

$$= x^3 - x^2x'_r - 2x^2x_p + 2xx_px'_r + x_p^2x - x_p^2x'_r$$

$$= x^3 - (x'r + 2x_p)x^2 + (2x_px'_r + x_p^2)x - x_p^2x'_r$$

Izjednačavanje koeficijenata:

$$k^2 = x'_r + 2x_p$$

$$x'_r = k^2 - 2x_p$$

$$= (\frac{3x_p^2 + A}{2y_p})^2 - 2x_p$$

Računanje koordinate y'_r sledi direktno iz jednačine prave:

$$y'_r = kx'_r + n$$

$$y'_r = \frac{3x_p^2 + A}{2y_p}x'_r + y_p - \frac{3x_p^2 + A}{2y_p}x_p$$

$$= \frac{3x_p^2 + A}{2y_p}(x'_r - x_p) + y_p$$

Konačno:

$$R(x_r, y_r) = (x_r, -y'_r)$$

VIŠESTRUKO SABIRANJE TAČKE ELIPTIČKE KRIVE

Sad, kada je definisano sabiranje i dupliranje tačke, moguće je višestruko sabiranje tačke. n-tostruko sabiranje tačke P samom sobom označava se sa nP.

Iako je moguće računati n-tostruku vrednost tačke kao uzastopno sabiranje tačke:

$$nP = P + P + P + \ldots + P = 2P + P + P + \ldots P$$

što zahteva n sabiranja, efikasniji pristup je korišćenje što više izračunatih dupliranih sabiraka.

Trostruka vrednost: $3P = P + P + P = 2P + P$, pri čemu je prvo sabiranje udvostručavanje tačke P dok je drugo sabiranje regularno sabiranje dve različite tačke ($2P$ i P)

Četvorostruka vrednost: $4P = P + P + P + P = 2P + 2P = 2(2P)$, pri čemu je jedno sabiranje upotrebljeno za računanje $2P$ i jedno sabiranje za $2P$ i $2P$

u opštem slučaju:

Parno n: $nP = P + P + P + \ldots + P = 2P + \ldots + 2P = 2(\frac{n}{2}P)$

Neparno n: $nP = P + P + P + \ldots + P = 2P + \ldots + 2P + P = 2(n-1)P + P$

Pri čemu je potrebno $O(log(n))$ množenja.

GRUPA TAČAKA ELIPTIČKE KRIVE

Skup tačaka eliptičke krive definisane na konačnom polju F_q sa prethodno definisanom operacijom sabiranja tačaka predstavlja komutativnu grupu. Oduzimanje se svodi na sabiranje inverzom drugog sabirka.

Neutral:	Inverz:	Asocijativnost
$P + 0 = P$	$P + (-P) = 0$	$(P + Q) + R = P + (Q + R)$
$0 + P = P$	$-P + P = 0$	
$0 + 0 = 0$		

Na osnovu ove osobine tačaka eliptičke krive mogu se izvesti oblici algoritama definisani u prethodnom poglavlju (nad grupom celih brojeva) sada nad grupom tačaka krive sa operacijom sabiranja tačaka. Napomena stoji da pojedine krive nemaju generator ili da imaju više generatora koji generišu različite delove krive, iz tog razloga je bitan pažljiv odabir parametara kako bi krive bile upotrebljive u realnim uslovima.

DIFFIE-HELLMAN NAD GRUPOM TAČAKA ELIPTIČKE KRIVE

Odabirom generatora krive za vrednost *g* i zamenom operacije stepenovanja (uzastopnog množenja) operacijom množenja (uzastopnog sabiranja), lako se konstruiše varijanta ovog algoritma nad eliptičkom krivom.

Postavka (tajna)

0. **Javni parametri:** *G, n,* gde je *gG* generator grupe tačaka eliptičke krive nad poljem *Fn* (***n* je prost broj**)

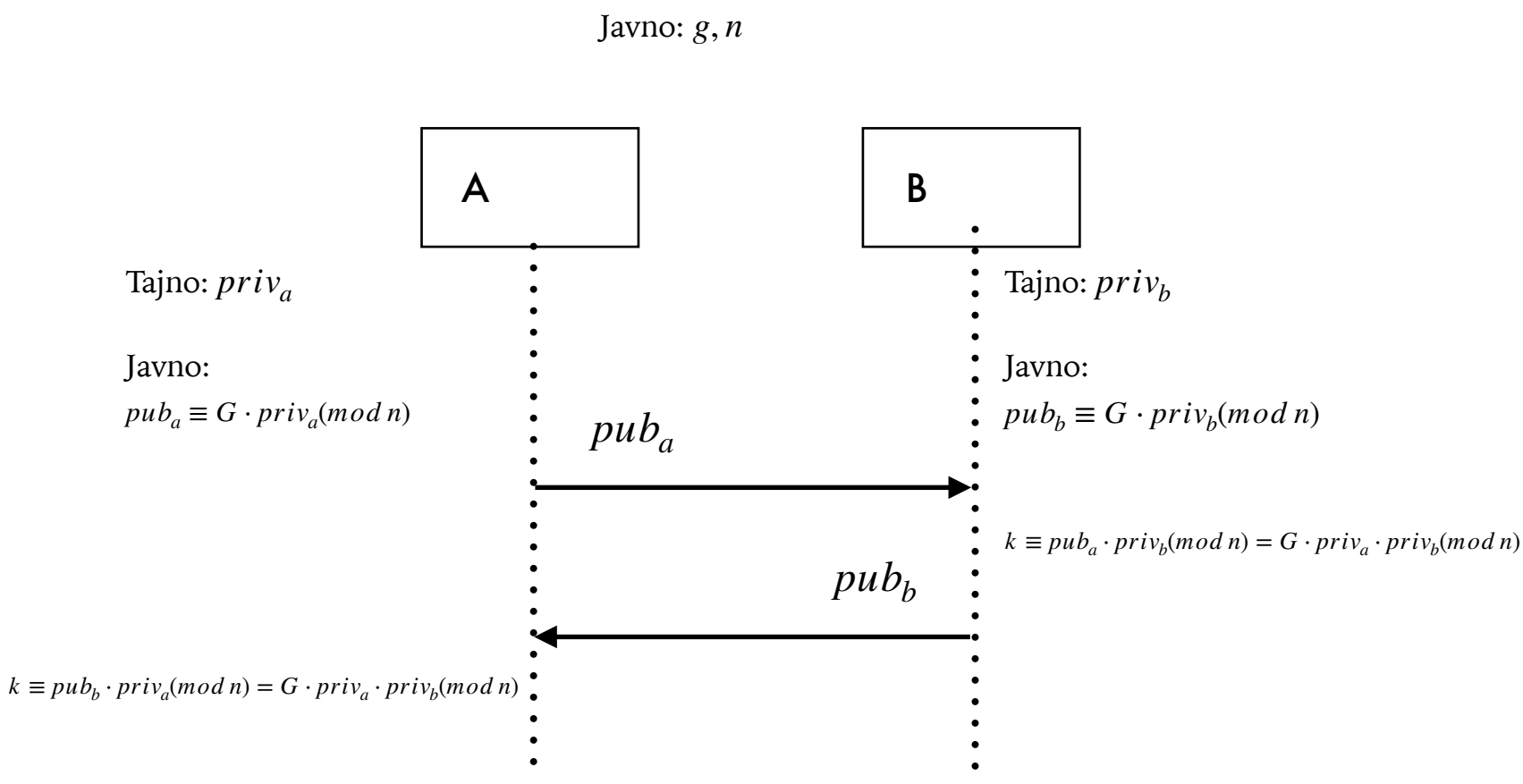
1. Učesnik A generiše privatni ključ pr_a kao slučajni broj iz intervala $(1,n)$
2. Učesnik B generiše privatni ključ pr_b kao slučajni broj iz intervala $(1,n)$
3. Učesnik A generiše javni ključ $pub_a = pr_a \cdot G(mod\ n) \equiv G + G + \ldots + G(mod\ n)$
4. Učesnik B generiše javni ključ $pub_b = pr_b \cdot G(mod\ n)$

Razmena ključa

1. Učesnik A šalje svoj javni ključ učesniku B
2. Učesnik B računa vrednost $k \equiv pub_a \cdot pr_b(mod\ n) \equiv G \cdot pr_a \cdot pr_b(mod\ n)$
3. Učesnik A šalje svoj javni ključ učesniku B
4. Učesnik B računa vrednost $k \equiv pub_b \cdot pr_a(mod\ n) \equiv G \cdot pr_a \cdot pr_b(mod\ n)$

Dobijena vrednost k predstavlja zajednički ključ za simetričnu enkripciju podataka koji će se razmenjivati između učesnika.

Diffie-Hellman Protokol nad eliptičkom krivom



Dijagram protokola razmene ključa Diffie-Hellman algoritmom

ELGAMAL NAD GRUPOM TAČAKA ELIPTIČKE KRIVE

ElGamal algoritam sa nad eliptičkom krivom konstruiše analogno *Diffie-Hemlan* algoritmu.

Postavka (tajna)

0. **Javni parametri:** g, n, gde je g generator grupe tačaka eliptičke krive nad poljem F_n (**n je prost broj**)

1. Učesnik A generiše privatni ključ pr_a kao slučajni broj iz intervala $(1,n)$

2. Učesnik B generiše privatni ključ pr_b kao slučajni broj iz intervala $(1,n)$

3. Učesnik A generiše javni ključ $pub_a = pr_a \cdot G(mod\, n) \equiv G + G + \ldots + G(mod\, n)$

4. Učesnik B generiše javni ključ $k \equiv pub_b \cdot pr_a(mod\, n) \equiv G \cdot pr_a \cdot pr_b(mod\, n)$

Enkripcija / dekripcija

1. Učesnik A generiše slučajni broj k iz intervala $(1,n)$
   $k = random(1,n)$

2. Učesnik A izračunava vrednost maskirajućeg ključa
   $mask_k \equiv G \cdot k(mod\, n)$

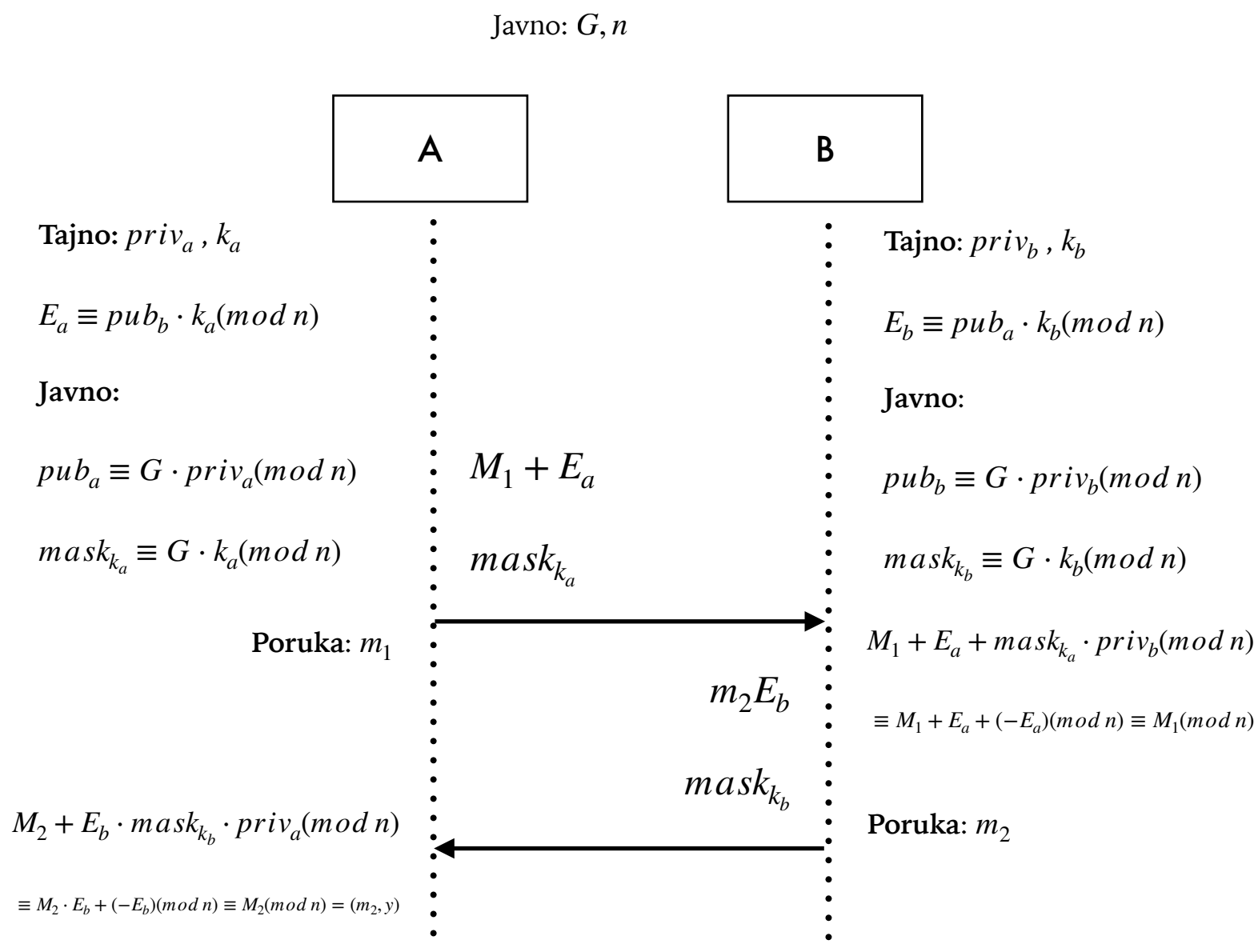
3. Učesnik A poruku m enkriptuje pomoću maskirajućeg i javnog ključa učesnika B. Prvo je potrebno poruku m mapirati na neku od tačaka krive oblika $M = (m, y)$, za neko y za koje tačka M pripada krivoj.
   $M_e \equiv M \cdot pub_b \cdot k(mod\, n) \equiv M + G \cdot priv_b \cdot k(mod\, n) \equiv M + G \cdot k \cdot priv_b(mod\, n) \equiv M + E(mod\, n)$

4. Učesnik A šalje enkriptovanu poruku m_e učesniku B, zajedno sa maskirajućim ključem
   $A - (M_e,\ mask_k) \rightarrow B$

5. Učesnik B izračunava vrednost E, pomoću maskirajućeg ključa i svog privatnog ključa
   $E \equiv mask_k \cdot priv_b(mod\, n) \equiv (G \cdot k) \cdot priv_b(mod\, n) \equiv G \cdot priv_b \cdot k(mod\, n)$

6. Učesnik B izračunava multiplikativni inverz tačke E po u grupi krive i njime sabira maskiranu poruku dobijenu od učesnika A
   $M_e + (-E)(mod\, n) \equiv m + E + (-E)(mod\, n) \equiv M(mod\, n) = (m, y)$

ElGamal Protokol nad eliptičkom krivom



Dijagram protokola enkripcije ElGamal algoritmom