Turinys

Šnoro identifikacija – Nulinių žinių įrodymas

(angl. *Schnorr Identification: Zero Knowledge Proof* - ZKP)

Nulinių žinių įrodymas – tai būdas įrodyti teiginio teisingumą neatskleidžiant paties teiginio. Įrodinėtojas yra šalis, bandanti įrodyti teiginį, o tikrintojas yra atsakingas už teiginio patvirtinimą.

Nulinių žinių įrodymai pirmą kartą pasirodė 1985 m. straipsnyje „Interaktyvių įrodymų sistemų žinių sudėtingumas“, kuriame pateikiamas šiandien plačiai naudojamas nulinių žinių įrodymų apibrėžimas:

Nulinių žinių protokolas – tai metodas, kuriuo viena šalis (tikrintojas) gali įrodyti kitai šaliai (tikrintojui), kad kažkas yra teisinga, neatskleisdama jokios informacijos, išskyrus tai, kad tam tikras konkretus teiginys yra teisingas.

Viešųjų parametrų apskaičiavimas

Viešieji parametrai **PP**=(***p***, ***g***)

Apskritai sudėtinga užduotis rasti generatorius aibėje $\mathbf{Z}_p^* = \{1, 2, 3, \ldots, p-1\}$, tačiau naudojant stiprų pirminį ***p*** ir *Lagranžo teoremą grupės teorijoje*, generatorių $\mathbf{Z}_p^*$ galima rasti atsitiktine tvarka. Paieška laikoma užbaiga jei tenkinamos dvi sąlygos:

1. jeigu ***p*** ir ***q*** yra stiprūs pirminiai $p = 2 \cdot q + 1 \rightarrow q = (p-1)/2$;
2. jeigu visi $g \in \Gamma$, $g^q \neq 1$ **mod** ***p***; and $g^2 \neq 1$ **mod** ***p***. **Tik 40% skaičių yra generatoriai.**

Pavyzdinis generatoriaus radimas (***g*** didinamas po vienetą, kol *ans* nelygus 1 ir neviršija ***p***):

```
>>p=genstrongprime(28)
p = 187086587
>> isprime(p)
ans = 1
>> q=(p-1)/2
q = 93543293
>> isprime(q)
ans = 1
>> g=2
>> mod_exp(g,q,p)
ans = 187086586
>> g=3;
>> mod_exp(g,q,p)
ans = 1
>> g=4;
>> mod_exp(g,q,p)
```

```
ans = 1
>>p=genstrongprime(28)
p = 144668519
>> q=(p-1)/2
q = 72334259
>> g=2;
>> mod_exp(g,q,p)
ans = 1
>> g=7;
>> mod_exp(g,q,p)
ans = 144668518
```

```
>> p=genstrongprime(28)
p = 224013599
>> q=(p-1)/2
q = 112006799
>> g=111;
>> mod_exp(g,q,p)
ans = 224013598
```

Toliau naudosime ***p***=int64(187086587); ***g***=2.

Privataus ir viešo raktų poros apskaičiavimas

Raktų generavimas susideda iš šių žingsnių:

1. Sugeneruoti privatų raktą (**PR**) x, pasirenkant atsitiktinį skaičių x ir patikrinti ar tenkinama sąlyga $2 \leq x \leq p$:

   ```
   >> x =int64(randi(p-1))
   x = 152676803
   ```

   ```
   >> 2<=x & x<=p
   ans = 1
   ```

2. Apskaičiuoti viešą raktą (**VR**) $a = g^x \bmod p$:

   ```
   >> a=mod_exp(g,x,p)
   a = 110652081
   ```

3. Konkretaus subjekto privatus raktas **PR** = x = 152676803, viešas raktas **VR** = a = 110652081.

Toliau naudosime subjektui **Aldona** raktų porą x=int64(152676803); a=int64(110652081).

Šnoro identifikacija

Šnoro identifikavimas – tai kriptografinis identifikavimo metodas, kurį aprašė 1991 Klausas Piteris Šnoras (angl. *Claus-Peter Schnorr*). Ši sistema yra viena pirmųjų nulinio žinojimo protokolų, leidžianti įrodinėtojui patvirtinti, kad jis žino tam tikrą slaptą informaciją (pvz., diskrečiąją logaritmo reikšmę **x**, be būtinybės jos atskleisti.

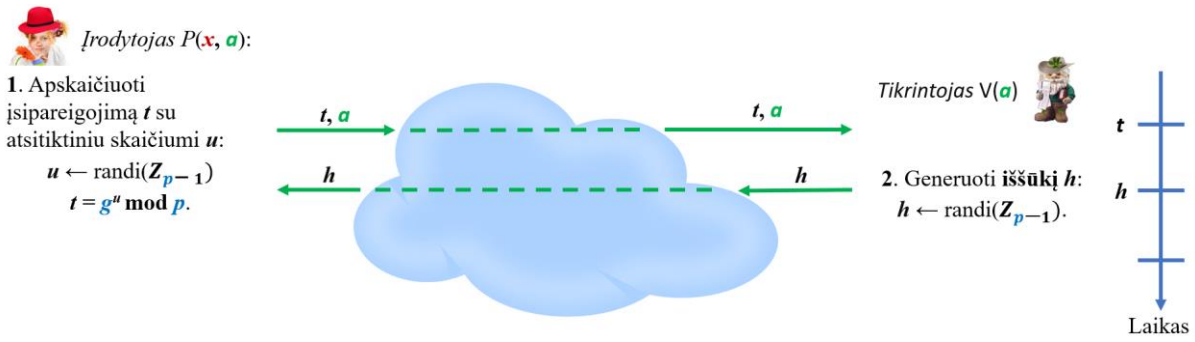
Šnoro identifikavimas[1] yra pagrįstas diskrečiojo logaritmo problemos sudėtingumu, užtikrinančiu jo saugumą. Dėl savo paprastumo ir efektyvumo šis metodas plačiai naudojamas skaitmeninėje tapatybės autentifikacijoje. Svarbiausias Šnoro identifikavimo privalumas yra tai, kad galima įrodyti tapatybės atributus ar kitą turimą slaptą informaciją, neatskleidžiant pačios informacijos ar papildomų duomenų.

Šnoro identifikacijos scenarijus: Aldona nori įrodyti Broniui, kad ji neatskleisdama žino savo privatųjį raktą $PR_A = x$, kurį atitinka viešasis raktas $VR_A = a = g^x \bmod p$.

Aldona mesdama iššūkį **Broniui**, sugeneruotą atsitiktinį slaptą parametrą u, kad tenkintų sąlygą $0 < u < p\text{-}1$ ir apskaičiuoja įsipareigojimą (angl. *commitment*) $t = g^u \bmod p$, kurį kartu su viešuoju raktu persiunčia **Broniui**.

Bronius atliepdamas į **Aldonos** įsipareigojimą sugeneruoja atsitiktinį skaičių h, kad tenkintų sąlygą $0 < h < p\text{-}1$ ir jį persiunčia **Aldonai**, kartu priimdamas jos iššūkį (angl. *challenge*), kad ji pagrįstai įrodys, jog iš tikro žino savo privatųjį raktą, kuris atitinka jos viešąjį raktą, jo neatskleisdama.

Supaprastinta įsipareigojimo ir iššūkio priėmimo apsikeitimo schema pateikiama 1 pav.



1 pav. Aldona įsipareigoja įrodyti, o Bronius priima iššūkį

Aldona

```
>> u=int64(randi(p-1))
u = 44901587
>> 0<u & u<p-1
ans = 1
>> t=mod_exp(g,u,p)
t = 86443002
```

Bronius

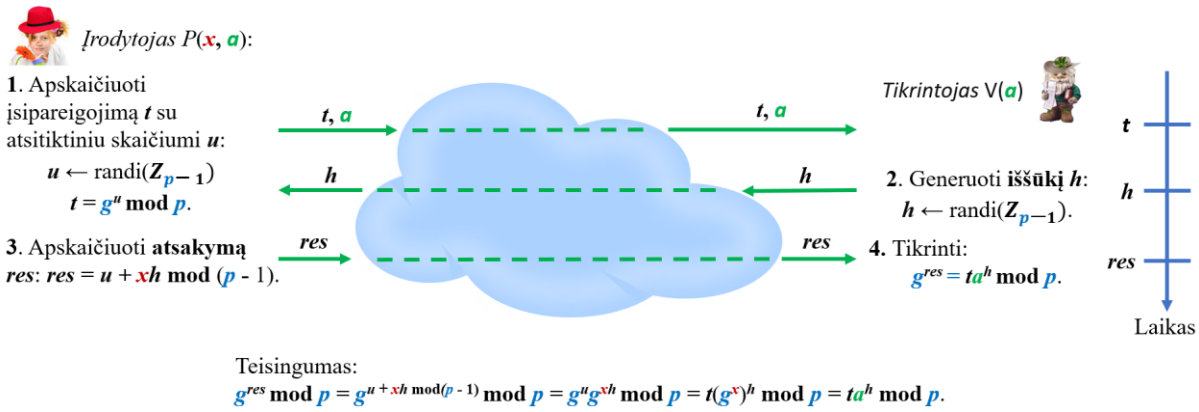
```
>> h=int64(randi(p-1))
h = 14978968
>> 0<h & h<p-1
ans = 1
```

[1] Alternatyvus šaltinis susipažinti su Šnoro identifikacija

Aldona reaguodama į tai, kad **Bronius** priėmė jos mestą iššūkį siunčia atsakymą (angl. *response*) $res = u + xh \bmod (p-1)$, kuris užbaigia įrodymų grandinę.

Taigi, **Bronius** žinodamas **Aldonos** įsipareigojimą, jos viešąjį raktą, bei galutinį atsakymą pasitikrina ir įsitikina, kad **Aldona** sako tiesą, jeigu tenkinama lygybė $g^{res} = ta^h \bmod p$ (V_1=V_2).

Supaprastinta Aldonos atsakymo ir Broniaus įsitikinimo schema pateikiama 2 pav.



Teisingumas:

$g^{res} \bmod p = g^{u+xh \bmod (p-1)} \bmod p = g^u g^{xh} \bmod p = t(g^x)^h \bmod p = ta^h \bmod p.$

2 pav. Aldonos atsakymas ir Broniaus įsitikinimas

Aldona

>> res=mod(u+x*h,p-1)
res = 176714471

Bronius

>> V1=mod_exp(g,res,p)
V1 = 153819493
>> a_h=mod_exp(a,h,p)
a_h = 121653689
>> V2=mod(t*a_h,p)
V2 = 153819493

>>**V1==V2**
ans = 1 ← jeigu **1 Aldona** įrodė **Broniui**

Bronius patiki, kad **Aldona** žino savo privatųjį raktą jo neatskleisdama, kurį atitinka jos viešasis raktas, tik tuomet, kai jos mėginimas įrodyti tenkina aukščiau pateiktą sąlygą.

Užduotys Šnoro identifikacijai.

Užduotims naudojami viešieji parametrai ***p***=int64(187086587); ***g***=2.

1. Turėdami **Aldonos** privatų raktą (**PR**$_A$) ***x*** ir atsitiktinį skaičių ***u***, **Broniaus** atsitiktinį skaičių ***h***, nustatykite, kuriam iš toliau esančių **Aldonos** atsakymų ***res*** apskaičiavimui, buvo panaudotos šios reikšmės:

   1. ***x***=int64(160531607), ***u***=int64(13163735), ***h***=int64(162930054);
   2. ***x***=int64(82952159), ***u***=int64(40053132), ***h***=int64(51284246);
   3. ***x***=int64(92714939), ***u***=int64(115680537), ***h***=int64(171057856);
   4. ***x***=int64(101079983), ***u***=int64(158245183), ***h***=int64(173420883).

   Atsakymai ***res***:

   1. ***res*** = 101036324;
   2. ***res*** = 43282537;
   3. ***res*** = 88003687;
   4. ***res*** = 56537196.

2. Turėdami **Aldonos** viešą raktą (**VR**$_A$) ***a*** ir įsipareigojimą ***t***, **Broniaus** atsitiktinį skaičių ***h***, **Aldonos** atsakymą ***res***, nustatykite, kuriame iš toliau pateiktų atvejų **Aldonai** pavyko įrodyti **Broniui**, kad ji žino savo privatųjį raktą jo neatskleisdama:

   1. ***a***=int64(75074745), ***t***=int64(42157078), ***h***=int64(13270500), ***res***=int64(106278411);
   2. ***a***=int64(110061633), ***t***=int64(10928254), ***h***=int64(46031156), ***res***=int64(171918286);
   3. ***a***=int64(42148327), ***t***=int64(96942883), ***h***=int64(92880168), ***res***=int64(129141525);
   4. ***a***=int64(177782970), ***t***=int64(131854891), ***h***=int64(179850472), ***res***=int64(67624064).

   Tik du kartus **Aldonai** pavyko įrodyti Broniui, kad ji žino savo privatųjį raktą jo neatskleisdama.

Šnoro parašas

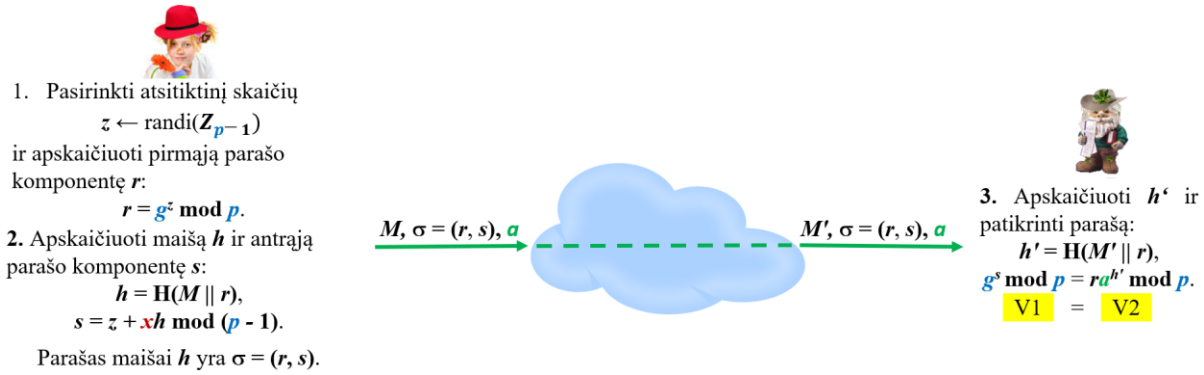
(angl. *Schnorr signature, S-Sig*)

Šnoro parašas – tai skaitmeninis parašas, kurio schemą 1991 pateikė Klausas Piteris Šnoras. Tai viena pirmųjų skaitmeninio parašo schemų, žinoma dėl savo paprastumo, kurios saugumas pagrįstas diskrečiojo logaritmo uždavinių neišsprendžiamumu. Šnoro parašas leidžia pasiekti mažesnį parašo dydį, greitesnį tikrinimo laiką. Svarbiausias Šnoro parašo privalumas yra raktų agregavimas – leidžia kelioms bendradarbiaujančioms šalims sukurti parašą, kuris galioja jų viešųjų raktų sumai.

Toliau naudosime tuos pačius viešuosius parametrus p=int64(187086587); g=2 ir viešo ir privataus raktų porą x=int64(152676803); a=int64(110652081), kaip ir skyriuje *Šnoro identifikacija – Nulinių žinių įrodymas*.

Pranešimas (tekstograma) $\boldsymbol{m}$:
>> m="Labas Broniau!"
m= Labas Broniau!

Supaprastinta parašo formavimo ir tikrinimo schema pateikiama 3 pav.



3 pav. Parašo formavimo ir jo patikrinimo schema

Aldona formuoja parašą σ pranešimui m su savo privačiuoju raktu (PR_A) x:

1. Sugeneruoti atsitiktinį skaičių $z \leftarrow \text{randi}(Z_{p-1})$, kad tenkintų sąlygą $1 < z < p-1$ ir apskaičiuoti pirmąją parašo komponentę $r = g^z \bmod p$:
   >> z=int64(randi(p-1))
   z = 16584813
   >> 1 < z & z<p-1
   ans = 1
   >> r=mod_exp(g,z,p)
   r = 168598583
2. Sujungti (angl. *concat*) m ir r, apskaičiuoti santrauką $h=H(m||r)$ ir apskaičiuoti antrąją parašo komponentę $s= z+xh \bmod (p-1)$:
   >> cn=concat(m,r)
   cn = Labas Broniau!168598583
   >> h=hd28(cn)
   h = 246269472
   >> s=mod((z+x*h),p-1)
   s = 167860211
3. Parašas h santraukai yra $\sigma = (r,s)$
   Pasirašyti(x, h) = $\sigma = (r,s)$ = (168598583, 167860211).
4. **Aldona** siunčia **Broniui** parašą σ, pranešimą m, savo viešą raktą a.

Bronius tikrina parašą **σ** pranešimui ***m***. Parašas **σ=(*r*,*s*)** pranešimui ***m*** yra patikrinamas naudojant **Aldonos** viešąjį raktą (VR_A) ***a***:

1. Sujungti (angl. *concat*) ***m*** ir ***r*** ir apskaičiuoti santrauką ***h'*=H(*m*||*r*)**:

   >> cn=concat(m,r)
   cn = Labas Broniau!168598583

   >> h=hd28(cn)
   h = 246269472

2. Apskaičiuoti $V_1=g^s \bmod p$ ir $V_2=ra^{h'} \bmod p$ ir patikrinti ar tenkinama lygybė $V_1=V_2$:

   V1

   >> V1=mod_exp(g,s,p)
   V1 = 71820516

   V2

   >> a_h=mod_exp(a,h,p)
   a_h = 167642616
   >> V2=mod(r*a_h,p)
   V2 = 71820516

   >> **V1==V2**
   ans = 1 ← jeigu **1** parašas tikras

Parašas galiojantis (***True***) jeigu: $PR_A = a = F(VR_A) = g^x \bmod p$; $h = h'$; $M = M'$.
Patikrinti$(a, \sigma, h') = P \in \{True, False\} \equiv \{1, 0\}$.

Tikrintojas **Bronius** priima parašą, jeigu parašas tenkina visas aukščiau pateiktas sąlygas, kitais atvejais parašą atmeta.

Užduotys Šnoro parašui.

Užduotims naudojami viešieji parametrai p=int64(187086587); g=2.

1. Turėdami privatų raktą (PR) x, atsitiktinį skaičių z, pranešimą m, nustatykite, kuriam iš pokalbio metu tarp **Aldonos** ir **Broniaus** toliau pateiktų parašų $\sigma = (r,s)$ formavimui buvo panaudotos šios reikšmės:

   1. x=int64(82952159), z=int64(72017296), m="Labas Broniau!";
   2. x=int64(160531607), z=int64(180040661), m="Labas Aldona!";
   3. x=int64(101079983), z=int64(66166232), m="Kada galėtume susitikti.";
   4. x=int64(92714939), z=int64(44531188), m="Susitikime vakare.".

   Parašai $\sigma = (r,s)$:

   1. r = 109813761, s = 124635270;
   2. r = 96115211, s = 24019457;
   3. r = 94843525, s = 124635270;
   4. r = 30351149, s = 24019457;
   5. r = 96115211, s = 23728252;
   6. r = 94843525, s = 130105774;
   7. r = 30351149, s = 23728252;
   8. r = 109813761, s = 130105774.

2. Turėdami viešą raktą (VR) a, pranešimą m, parašą $\sigma = (r,s)$, nustatykite, kuriems **Aldonos** ir **Broniaus** pranešimams suformuoti parašai yra galiojantys, panaudojant šias reikšmes:

   1. a=int64(75074745), m = "Šalia seno ąžuolo.", r=int64(114423569); s=int64(153854495);
   2. a=int64(177782970), m = "Iki pasimatymo.", r=int64(15266365); s=int64(177172395);
   3. a=int64(42148327), m = "Iki greito.", r=int64(173005895); s=int64(1497974);
   4. a=int64(110061633), m = "Šalia didelio kelmo.", r=int64(102901505); s=int64(73082431).

   Tik du parašai $\sigma = (r,s)$ galioja.

3. Turėdami viešą raktą (VR) a ir parašą $\sigma = (r,s)$, nustatykite, kuriems **Aldonos** ir **Broniaus** pranešimams m buvo suformuotas parašas, panaudojant pateiktas reikšmes:

   1. a=int64(75074745), r=int64(107423137), s=int64(68853422);
   2. a=int64(110061633), r=int64(60182491), s=int64(154952475);
   3. a=int64(42148327), r=int64(137845968), s=int64(133417557);
   4. a=int64(177782970), r=int64(145654127), s= int64(114224801).

   Pranešimai m:

   1. m_1 = "Kelintą valandą vakare.";
   2. m_2 = "19 valandą.";
   3. m_3 = "Kurioje vietoje.";
   4. m_4 = "Jaukioje parko kavinėje.".

Ronaldas Rivestas, Adi Shamiras ir Leonardas Adlemanas

(angl. *Rivest–Shamir–Adleman, RSA*)

RSA – viena labiausiai paplitusių viešojo rakto kriptografinių sistemų, kurią 1977 m. Masačusetso technologijos institute (MIT) sukūrė Ronaldas Rivestas, Adi Shamiras ir Leonardas Adlemanas. 17 metų RSA sistema buvo saugoma JAV patento, tačiau 2000 m. rugsėjo mėnesį šio patento galiojimas baigėsi ir nuo tada RSA sistemą galima naudoti laisvai.

RSA asimetrinės kriptografinės sistemos privalumas tas, kad ji gali būti naudojama ir asimetrinio šifravimo, ir elektroninio parašo sistemose.

Viešojo ir privataus rakto apskaičiavimas

RSA sistemos pagrindas yra trys tarpusavyje susiję skaičiai. Du iš jų yra visiems žinomi ir sudaro viešąjį raktą **VR** = (**n**, **e**), trečiasis yra slaptas **PR** = (***d***) ir žinomas tiktai rakto savininkui. Raktų generavimas susideda iš šių žingsnių:

1. Sugeneruoti du pakankamai didelius pirminius skaičius ***p*** ir ***q***, kurie turi būti ***p*** ≠ ***q***:
   >> p=genprime(15) >> q=genprime(15)
   p = 18911 q = 17027
2. Apskaičiuoti sandaugą **n** = ***pq***. Ši sandauga yra vienas iš viešo rakto parametrų:
   >> n=int64(p*q)
   n = 321997597
3. Apskaičiuoti Eulerio funkciją φ(**n**), kai ***p*** ir ***q*** yra pirminiai, tai φ(**n**) = (***p*** – 1)(***q*** – 1):
   >> fy=int64((p-1)*(q-1))
   fy = 321961660
4. Parinkti tokį sveikąjį skaičių **e** (1 < **e** < φ(**n**)), kad **e** ir φ(**n**) būtų reliatyviai pirminiai skaičiai, t. y. **e** ir φ(**n**) ir bendras didžiausias daliklis būtų 1. **e** yra antrasis viešo rakto parametras:
   >> e=genprime(14) 1 < e < fy gcd(e, fy)
   e = 17977 ans = 1 ans = 1
5. Rasti privatų raktą/slaptąjį parametrą, kuris dažniausiai randamas naudojant išplėstinį Euklido algoritmą ***d*** = **e**$^{-1}$ **mod** φ(**n**), kad **e*d*** **mod** φ(**n**) = 1:
   >> d=mulinv(e,fy) ← geriau už eeuklid() >> mod(e*d,fy)
   d = 204277393 ans = 1
6. Konkretaus subjekto (toliau bus Aldonos) RSA raktų pora yra **PR** = (***d***) ir **VR** = (**n**, **e**), **PR** = (***d***) = (204277393) ir **VR** = (**n**, **e**) = (321997597, 17977)

RSA saugumas remiasi tuo, kad turint tik **n**, ***p*** ir ***q***, jų atkūrimas per priimtiną laiką yra praktiškai neįmanomas. Šiuo metu rekomenduojamas minimalus raktų ilgis yra 2048 bitai.

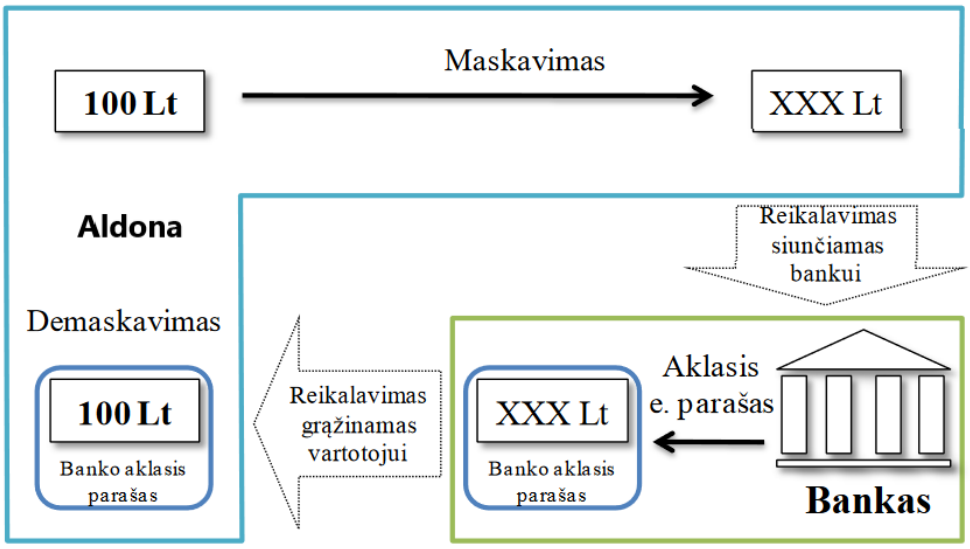
RSA Aklojo e. parašo formavimas ir tikrinimas

(angl. *RSA Blind Signature*)

Aklasis elektroninis parašas skiriasi nuo tradicinių parašų tuo, kad pasirašoma užmaskuota tikroji vertė pvz., 100 Lt užmaskuojama kaip 5346, kuriai suformuojamas parašas **ϭ′**. Atlikus demaskavimo veiksmą, gaunamas parašas **ϭ** tikrajai vertei pvz., 100 Lt, tarsi vertė nebūtų buvusi užmaskuota. Tai užtikrina vertės autentiškumą, kartu išlaikant jos konfidencialumą.

RSA kriptosistemos pagrindu 1982 m. Deivido Ly Čiaumo (angl. *David Lee Chaum*) pasiūlyta pirmoji aklojo e. parašo schema (žr. 4 pav.):

1. Aldona sukuria reikalavimą e. pinigui gauti ir jį taip užmaskuoja, kad niekas kitas negalėtų sužinoti reikalaujamo e. pinigo nominalo;
2. Bankas pasirašo užmaskuotą e. pinigą;
3. Aldona demaskuoja banko pasirašytą e. pinigą, tačiau išsaugo autentišką banko parašą ant e. pinigo nominalo.
4. Aldonai išleidžiant e. pinigus, pardavėjas ar kitas paslaugų teikėjas patikrina jų galiojimą, kreipdamasis į banką. Bankas patikrina e. pinigo parašą ir leidžia atlikti mokėjimą, jei parašas yra galiojantis.



4 pav. Aklojo e. parašo schema

Toliau naudosime subjektui **Bankas** RSA raktų porą **PR** = (***d***) = (204277393) ir **VR** = (**n**, **e**) = (321997597, 17977):

```
>> d=int64(204277393); n=int64(321997597); e=17977;
```

Maskuojama piniginė vertė Π=***p*** (e. pinigo nominalas Π, pvz. Π = 100 (tarkime100 Lt)) turi tenkinti sąlygą 1 < ***p*** < **n**:

```
>> p=100
p = 100
```

```
>> 1<p & p<n
ans = 1
```

Aldona su banko viešuoju raktu **VR** = (**n**, **e**) užmaskuoja piniginę vertę Π:

1. Atsitiktinai sugeneruoti skaičių **r**, kad **r** ir **n** bendras didžiausias daliklis būtų 1 ir $1 \leq r \leq n$:

```
>> r=int64(randi(n))        >> gcd(r,n)        >> 1<=r & r<= n
r = 61584152              ans = 1            ans = 1
```

2. Užmaskuoti Π, apskaičiuojant reikšmę $\Pi' = \Pi \cdot r^e \bmod n$; ***p_mask***=Π′:

```
>> r_e=mod_exp(r,e,n)                >> p_mask=mod(p*r_e,n)
r_e = 197680157                   p_mask = 126162283
```

3. **Aldonos** užmaskuota piniginė vertė Π′ piniginei vertei Π
   Maskuoti(**e**, **n**, Π) = Π′ = (126162283).
4. **Aldona** siunčia **Bankui** užmaskuotą piniginę vertę Π′

Bankas pasirašo ant užmaskuotos piniginės vertės Π′ su savo privačiuoju raktu **PR** = (***d***):

1. Pasirašyti Π′, suformuojant parašą $\sigma' = (s') = (\Pi')^d \bmod n$:

```
>> s_mask=mod_exp(p_mask,d,n)
s_mask = 124046105
```

2. **Banko** parašas piniginei vertei Π′ yra **σ′ = (*s′*)**
   Pasirašyti(***d***, **n**, Π′) = **σ′ = (*s′*)** = (124046105).
3. **Bankas** siunčia **Aldonai** parašą **σ′**.

Aldona demaskuoja parašą **σ′ = (*s′*)** apskaičiuodama **σ = (*s*)** su ankščiau sugeneruotu atsitiktiniu skaičiumi **r**:

1. Apskaičiuoti $\mathbf{r}^{-1} \bmod \mathbf{n}$:

```
>> r_m1=mulinv(r, n)
r_m1 = 174025693
```

2. Apskaičiuoti $\boldsymbol{s} = \boldsymbol{s'} \cdot \mathbf{r}^{-1} \bmod \mathbf{n}$:

```
>> s=mod(s_mask*r_m1,n)
s = 81355534
```

3. Demaskuotas banko aklasis parašas **σ = (*s*)** piniginei vertei Π:
   Demaskuoti (**r**, *s′*) = **σ = (*s*)** = (81355534).
4. **Aldona** atėjus laikui, t.y norėdama išleisti pasirašytą piniginę vertę Π, **Broniui** pardavėjui siunčia Π ir demaskuotą parašą **σ=(*s*)**.

Bronius pardavėjas tikrinimą paveda atlikti **Bankui**, kuris naudodamasis savo viešuoju raktu tikrina parašą **σ=(*s*)** piniginei vertei Π:
VR = (**n**, **e**):

1. Apskaičiuoti $\mathbf{V} = s^e \bmod \mathbf{n}$ ir patikrinti ar tenkinama lygybė **V**=Π:

```
>> V=mod_exp(s,e,n)
V = 100
```

```
>> V==p
ans = 1 ← jeigu 1 parašas tikras
```

Parašas galiojantis (***True***) jeigu: **VR** = (**n**, **e**); V = Π.
Patikrinti(**VR**, **σ**, Π) = **P**∈{***True***, ***False***}≡{**1**, **0**}.

Bronius pardavėjas parduoda prekę, jeigu gaunamas patvirtinimas iš banko, kad parašas galioja t.y. tenkinama aukščiau pateiktą tikrinimo sąlyga. Kitais atvejais prekės neparduoda.

Užduotys Aklajam e. parašui.

Užduotims naudokite Banko RSA raktų porą.

1. Turėdami piniginę vertę $\Pi=p$ ir atsitiktinį skaičių r, nustatykite, kuriai iš **Aldonos** užmaskuotų piniginių verčių **Bankas** suformavo parašą $\sigma'=(s')$, naudojant šias reikšmes:
   1. p=342, r=int64(139023994);
   2. p=6, r=int64(196405921);
   3. p=1652, r=int64(78850462);
   4. p=15, r=int64(218295098).

   Parašai $\sigma'=(s')$:
   1. s' = 289995150;
   2. s' = 316951343;
   3. s' = 265945455;
   4. s' = 314294550.
2. Turėdami parašą $\sigma'=(s')$ ir atsitiktinį skaičių r, nustatykite kuriam demaskuotam parašui ir **Aldonos** išleidžiamai piniginei vertei **Bankas** yra suformavęs parašą, naudojant šias reikšmes:
   1. s_mask=int64(58364639), r=int64(308550421);
   2. s_mask=int64(175178662), r=int64(271169128);
   3. s_mask=int64(299273336), r=int64(276385525);
   4. s_mask=int64(235475588), r=int64(306660562).

   Demaskuoti parašai $\sigma=(s)$ ir piniginės vertės p:
   1. s = 115177084, p = 15;
   2. s = 108072303, p = 15;
   3. s = 316188901, p = 1652;
   4. s = 224281475, p = 1652;
   5. s = 316188901, p = 6;
   6. s = 224281475, p = 6;
   7. s = 115177084, p = 342;
   8. s = 108072303, p = 342.