Miselni poker

Miha Štravs

August 2020

1 Povzetek

Problem miselnega pokra je tema, ki se jo že dolgo raziskuje. Trenutno se za igranje pokra preko spleta uporablja centralizirane protokole s centrom zaupanja. Vse več pa je predlogov za decentralizirane protokole, ki se znebijo slabih lastnosti uporabe centra zaupanja. Kljub temu za enkrat še ni takih protokolov, ki bi dosegli dovolj dobro hitrost pri željeni varnosti. Naloga se osredotoči na protokol Kaleidoscope, ki je prvi s formalnim dokazom varnosti in hitrostjo primerljivo z ostalimi hitrejšimi protokoli za miselni poker.

2 Uvod

Poker je ena izmed bolj popularnih iger na srečo. Je igra s kartami, kjer igralci stavijo na kombinacijo svojih kart v roki. Že v osemdesetih letih so se raziskovalci začeli spraševati, če je možno preko komunikacijskega kanala (takrat je bilo to mišljeno preko telefona) opraviti pošteno igro pokra (Shamir et al., 1981). Za ta prvi poiskus se je kasneje našlo par varnostnih napak (Lipton, 1981; Coppersmith, 1985). S prihod interneta je ta tema postala bolj popularna, ker je z njim prišla želja po igrah na srečo preko spleta. Članek (Barnett and Smart, 2003) je predstavil metodo, ki je bolj unčikovita od prvotne. Tisto leto je izšla tudi metoda podobna prejšnji (Castella-Roca et al., 2003), ki uporablja homomorfno šifriranje. Kasneje so odkrili še metodo za varno in hitro mešanje kart (Bayer and Groth, 2012), ki se lahko uporabi v prejšnih dveh metodah. V zadnjem desetletju se je pojavilo vprašanje kako ustvariti pretok denarja med igrlaci. Zaradi varnosti in priročnosti večina raziskovalcev išče rešitve v uporabi *veriženih skupkov* (angl. blockchain) (Kumaresan et al., 2015).

Večina trenutnih protokolov za igranje pokra preko spleta v uporabi je centraliziranih z uporabo centra zaupanja. To pomeni, da se igralci povežejo na center zaupanja in upajo, da bo ta deloval pravično in po pravilih. Centeri zaupanja pa imajo par slabosti. So šibka točka napadov. Torej, če nam ga uspe napasti lahko pridobimo podatke o igri in jih po možnosti spreminjamo v našo korist. Če je ta center zaupanja zloben, nas lahko finančno oškoduje. Poleg teh dveh slabosti pa moramo uporabo centra ponavadi plačevati. Alternativa so protokoli za decentralizirano igro pokra preko spleta z uporabo protokolov

za *varno več-osebno računanje* (ang. Safe multiparty computation). Z odstranitvijo centra zaupanja se znebimo prejšnih problemov in pridobimo par novih. Ob pošteni igri pokra vsak igralec vidi svoje karte, karte na mizi in karte tistih, ki jih pokažejo na koncu igre. Če igralec med igro odvrže karte, njihove vsebine nikoli ne izvemo. Kako ugotoviti, če ostali igralci igrajo pošteno, brez tega, da bi poznali celotno stanje igre? To se razreši z uporabo dokazov brez razkritja znanja. Igralcu postavimo problem, ki ga zna razrešiti pravilno, če igra pošteno. Če igra nepošteno pa lahko odgovori pravilno z ugibanjem z neko majhno verjetnostjo. Če postavimo ta problem večkrat in igralec vsakič odgovori pravilno je verjetnost goljufije poljubno majhna. Naslednji problem je to, da lahko igralec sredi igre preneha igrati in se zaradi neaktivnosti tega igralca igra ne more nadaljevati. Vprašanje je tudi, kako bi implementirali varen večkraten prenos denarja med igralci. Ker nimamo centra, ki bi denar hranil, potrebujemo konstanten pretok denarja med igralci. To je razlog, da večina implementacij takih protokolov, ki podpirajo igro z denarjem, uporabi *verižene skupke*, za lažjo implementacijo in uporabo. Velik problem člankov na temo miselnega pokra je tudi to, da jih veliko nima formalnega dokaza za varnost in se je že zgodilo, da se je kasneje dokazalo, da niso varni. Na primer članek (Castella-Roca and Sebé, 2005) je naslovil problem izhoda sredi igre. Vendar pa se je kasneje izkazalo, da je imela njihova rešitev varnostno napako, ki je omogočala igralcem, da so lahko vlekli določene karte z večjo verjetnostjo. V tem članku se predstavi protokol Kaleidoscope (David et al., 2018), ki se spopade z vsemi naštetimi problemi. Poleg tega pa doseže časovno kompleksnost, ki je primerna za uporabo.

V naslednjem poglavju so opisane zahteve protokola in kakšen doprinos ima ta protokol pri razvoju decentraliziranih protokolov za miselni poker. Nato se predstavi znanja uporabljena za implementacijo protokola. Sledi opis igre pokra in nato še opis poteka z uporabo protokola Kaleidoscope. Temu sledi poglavje s kratkim opisom dokaza varnosti protokola in zahtevami, da je protokol varen.

3 Zahteve

Pri ustavrjanju protokola so si avtorji podali štiri zahteve, ki jih mora protokol izpolnjevati:

- **Hitrost**: Protokol mora biti dovolj hiter, da se primerja z ostalimi protokoli, ki so dovolj hitri za uporabo.
- **Varnost**: Protokol potrebuje formalni dokaz varnosti.
- **Kazni**: Če je kateri od igralcev neaktiven preveč časa, ali pa, da se ugotovi, da ne igra po pravilih, se ga finančno kaznuje.
- **Nagrade**: Po končanem krogu igre mora biti denar varno razdeljen med ostalimi igralci.

Predlagan protokol je prvi, ki dosega vse naštete lastnosti. V člankih (Bentov and Miller, 2017; Kumaresan et al., 2015) je avtorjem sicer uspelo priti blizu tem

pogojem ampak v (Bentov and Miller, 2017) ni formalnega dokaza varnosti in metode v (Kumaresan et al., 2015) so prepočasne, da bi izpolnilne prvi pogoj. Kljub temu pa so se avtorji zgledovali po teh dveh člankih, ko so ustvarjali protokol Kaleidoscope.

Cilj je ustvariti hiter protokol za igranje pokra, ki ima formalni dokaz varnosti in zmanjšanje količine podatkov ter uporabe *veriženih skupkov*, ki se jih uporablja za varno razporejanje denarja igralcev. Ker se želi robusten protokol, se vsi neaktivni igralci in igralci, ki pošiljajo napačne podatke finančno kaznujejo. Hitrost pa se pridobi s tem, ko zmanjšamo število dostopov na *veriženih skupkih*. Do teh se dostopa le ob koncu kroga, ko zmagovalec pobere zastavljen denar, ali pa takrat, ko se kaznuje igralca.

4 Uporabljena znanja

V tem poglavju se predstavi znanja, ki se jih uporabi pri implementaciji metode. Pri razlagi neinteraktivnih dokazov za lažje razumevanje predstavimo postopek dokazovanja. Pri tem uporabimo preverjevalca Petra in dokazovalca Domna. Za neinteraktivne dokaze in ElGamalov kriptosistem se uporabi števila iz primerne grupe G. Družina primernih grup je opisana v poglavju 7.

4.1 Neinteraktiven dokaz znanja za enakost dveh diskretnih logaritmov

Domen dokazuje, da pozna skrito število a. Javno znana števila so x, y, g, h, kjer drži $x = g^a$ in $y = h^a$. Domen naključno izbere vrednost w in pošlje Petru $a_1 = g^w$ in $a_2 = h^w$. Peter nato izbere naključno vrednost e in jo pošlje dokazovalcu. Domen nato odgovori s številom $z = w - ae$. Če $a_1 = g^z x^e$ in $a_2 = h^z y^e$ Peter sprejme njegov dokaz, da pozna skrito število. Dokaz se krajše zapiše kot DLEQ(g, x, h, y) (**D**iscrete **L**ogarithm **EQ**ual).

4.2 Neinteraktiven dokaz znanja diskrenega logaritema

Preprostejša različica, kjer ima Domen skrivno število a in javni ključ $x = g^a$. Domen uporabi le $a_1 = g^w$. Ko Peter določi naključno vrednost e in dokazovalec izračuna $z = w - ae$ se preveri $a1 = g^z x^e$. Dokaz krajše zapišemo kot DLOG(g, x) (**D**iscrete **LOG**arithm).

4.3 ElGamalov Kriptosistem z (n, n)-mejo

Kriptosistem z mejo (t, n) dovoli, da n udeležencev zgenerira javni ključ s katerim se lahko zašifrira besedilo tako, da se za dešifriranje potrebuje vsaj t udeležencev. V Kaleidoscope protokolu se uporabi ElGamalov Kriptosistem z (n, n) mejo pri komunikaciji n igralcev. Uporabi se multiplikativna grupa G, ki zagotovi težavnost Diffie-Hellman odločitvenega problema. Družina takih grup je opisana v poglavju 7. Grupa G ima generator g. Najprej si vsak igralec P_i

naključno izbere skriti ključ s_i. Nato izračuna še del javnega ključa $h_i = g^{s_i}$. Tega nato deli z ostalimi skupaj z dokazom DLOG(g, h_i), da dokaže svoje znanje skritega ključa s_i. Javni ključ nato vsi sestavijo tako, da vse deleže zmnožijo $h = \prod_{i=1}^{n} h_i = g^{\sum_{i=1}^{n} s_i}$. Skupaj si torej delijo skriti ključ $\sum_{i=1}^{n} s_i$, ne da bi ga kdo od igralcev poznal v celoti.

Čistopis $m \in G$ se šifrira s pomočjo javnega ključa h. Najprej se naključno izbere $r \in G$. Nato pa zašifriramo sporočilo v par $c = (c_1, c_2) = (g^r, h^r m)$

Če želimo spremeniti tajnopis, ne da bi spremenili vsebine si lahko naključno izberemo nek $k \in G$. Če je šifrirano besedilo $c = (c_1, c_2)$, je spremenjeno besedilo $c' = (g^k c_1, h^k c_2)$.

Pri dešifriranju igralec P_i pošlje c_1 del tajnopisa ostalim udeležencem. Igralec P_j mu vrne $d_j = c_1^{s_j}$ in dokaz DLEQ(g, h_j, c_1, d_j), da je d_j pravilen. Igralec P_i nato izračuna $d_i = c_1^{s_i}$. Udeleženec P_i lahko nato izračuna čistopis po formuli.

$$\frac{c_2}{\prod_{i=1}^{n} d_i} = \frac{c_2}{c_1^{\sum_{i=1}^{n} s_i}} = \frac{m(g^{\sum_{i=1}^{n} s_i})^r}{g^{r \sum_{i=1}^{n} s_i}} = m.$$

4.4 Neinteraktiven dokaz pravilnega mešanja

Imamo 52 zašifriranih kart v ElGamalovem kriptosistemu. Karte nato premešamo s permutacijo in uporabimo lastnost, da lahko spremenimo šifrirano sporočilo brez, da bi spremenili vsebine ali ključev. Po permutaciji in spremembi sporočil želimo pokazati, da smo to naredili na pošten način. Hočemo dokazati, da poznamo permutacijo in število k, ki je bilo uporabljeno za spremembo sporočil. Dokaz krajše zapišemo kot ZKSH(π, k, c ,c') (**Z**ero-**K**nowledge proof of a **SH**uffle), kjer je π permutacija, k naključno število za spremembo tajnopisev, c originalen seznam šifriranih kart, c' pa seznam kart po mešanju in spremembi tajnopisov. Dokaz je bolj podrobno opisan v (Bayer and Groth, 2012).

5 Potek igre

Obstaja več vrst pokra. Trenutno najbolj igrana različica je Texas Hold'em. Za lažje razumevanje delovanja protokola bo najprej predstavljen potek igre Texas Hold'em pokra.

Na začetku igre prinese vsak igralec denar s katerim bo stavil. Igralci imajo lahko med seboj različno začetno količino denarja. Na začetku se tudi dogovorijo kateri od igralcev bo začel z majhno začetno stavo, velikost majhne začetne stave ter krožni vrstni red igralcev za potek igre. Za igranje igre se uporabi 52 kart. Karte se delijo na ♠, ♢, ♣ in ♡. Za vsak znak imamo številke od 2 do 10 ter fanta, damo, kralja in as.

Karte se premeša in razdeli med igralce. Vsak igralec dobi dve karti. Igralec z začetno majhno stavo mora nujno staviti velikost majhne stave igralec za njim pa velikost velike stave, ki je enkrat večja od majhne. Nato poteka stavljenje v prej dogovorjenem krožnem vrstnem redu začenši z igralcem, ki je na vrsti za veliko začetno stavo. Igralec na vrsti ima pet možnosti:

- **Fold**: Igralec odvrže karte. Igralec se s tem odpove zmagi in ne stavi več. Če je že stavil denar bo ta denar šel k zmagovalcu. Ostali igralci nikoli ne izvejo katere karte je igralec imel.
- **Check**: V primeru, da je igralec že zastavil toliko kot je trenutna največja stava lahko stavi nič in ostane v igri.
- **Call**: Igralec zviša svojo stavo na trenutno največjo stavo.
- **Raise**: Igralec poviša prejšno najvišjo stavo.
- **All-in**: Igralec stavi celoten denar s katerim igra. Igralec je v igri do konca in ne stavi več. Če kateri od ostalih igralcev stavi nad vrednostjo stave igranega **All-ina** se višek obravnava posebej. V primeru, da igralec z **All-inom** zmaga pripada višek tistemu z najboljšimi kartami izmed preostalih igralcev.

Ko poteče točno en tak krog stav, da noben izmed igralcev ne viša najvišje trenutne stave, se stavljenje konča.

Iz kupa premešanih kart se nato na mizo postavi tri karte tako, da jih vidijo vsi igralci. Za tem se zopet začne stavljenje. Začne tisti, ki je imel na začetku obvezno majhno stavo. Nato se še dvakrat položi ena karta iz kupa na mizo in za vsako položeno karto se zopet stavi z istim vrstnim redom. Na koncu igralci pokažejo svoje karte. Začne tisti, ki je imel začetno obvezno majhno stavo in nato se nadaljuje v dogovorjenem krožnem vrstnem redu. Vsak igralec lahko pokaže karte ali pa izvede **Fold** in se odpove možnosti zmage v zameno za tajnost njegovih kart. Na koncu zmaga tisti z najboljšo kombinacijo kart, ki se jim ni odpovedal z **Fold** potezo. V primeru, da se kadarkoli med igro zgodi, da je samo še en igralec, ki se ni odpovedal svojim kartam, ta igralec zmaga. V tem primeru ne pokaže svojih kart. Naslednji krog igre ostane enak krožni vrstni red obvezna začetna majhna in velika stava se pa pomakne na naslednjega igralca po dogovorjenem vrstnem redu.

6 Protokol

Potek igre delimo na dva dela. Prvi del je funkcionalnost F, ki se izvaja na *verižnih skupkih*. Procesi, ki se izvajajo na *verižnih skupkih*, so varni ampak tudi počasni. Večino igre se zato izvaja preko komunikacije preko ElGamalovega kriptosistema.

6.1 Funkcionalnost F

Funkcionalnost F je del, kjer se zgodi prenos denarja. Stanje igre se pridobi od igralcev preko t. i. prič. **Priče** so shranjena podpisana sporočila vseh igralcev o uspešno izvedeni potezi. Igralci za podpis uporabijo svoj delež zasebnega ključa iz ElGamalovega kriptositema.

Igralci si delijo funkcionalnost F, ki vsebuje parametre igre. Funkcionalnost vsebuje podatke o majhni in veliki začetni obvezni stavi ssb, sbb, denar igralca ob začetku igre t, varnostni depozit d, velikost kompenzacije ob goljufiji q in časovno omejitev ℓ. Te parametri predstavljajo podatke dogovorjene pred začetkom izvedbe protokola. Varnostni depozit d mora biti vsaj $(n-1)q$.

6.1.1 Vpis v igro

Ob začetku protokola pošlje vsak igralec P_i denar $t+d$, svoj del javnega ključa h_i, ključ za preverjanje podpisa igralca V_i in dokaz o znanju deleža skritega ključa DLOG(g, h_i). Če kateri od igralcev tega ne stori v časovni omejitvi ℓ, imajo ostali igralci možnost zapustiti igro. Če kateri izmed igralcev zapusti igro, se mu vrne njegov vplačan denar.

6.1.2 Izpis iz igre

Ob koncu kroga imajo igralci možnost prenehanja igranja. Igralec pošlje sporočilo, da želi prenehati skupaj s trenutnim stanjem denarja in podpisom vseh ostalih igralcev, da se strinjajo z njegovimi podatki o trenutnem stanju igre. Igralec nato dobi izplačano njegovo stanje in varnostni depozit, ki ga je dal ob začetku igre.

6.1.3 Obnovitev

Če gre kaj narobe, lahko kateri koli od igralcev kadar koli pošlje zahtevo po obnovitvi. V zahtevo doda zadnjo shranjeno pravilno potezo s podpisi vseh igralcev, ki potrjujejo pravilnost koraka in svoje trenutno stanje. Za tem morajo to storiti tudi vsi ostali igralci. Če ima kateri od igralcev zastarelo zadnjo shranjeno pravilno potezo dobi informacije o novejših potezah od soigralcev. Če se vidi, da je kateri od igralcev dajal napačne informacije oziroma ni storil poteze v določenem času se mu vzame denar iz njegovega varnostnega depozita in ga razdeli med ostale igralce tako, da vsak dobi q denarja. Vsak od igralcev nato dobi trenuten denar in denar, ki ga je stavil v trenutni igri. Ko je denar izplačan, se igra takoj konča.

6.2 Komunikacija med igralci

6.2.1 Prijava igralcev v igro

Igralci na začetku zgenerirajo javni ključ in deleže skritega ključa za ElGamalov kriptosistem z mejo (n, n) za ciklično grupo G z generatorjem g. Vsak igralec P_i ima skriti delež ključa s_i. Vsi si pa delijo javni ključ h ter vse njegove deleže h_i. Igralci se vpišejo v igro preko funkcionalnosti F. Vsakemu igralcu se nato dodeli njegovo mesto v vrstnem redu igre. Inicializira se tudi vektor stav, ki je na začetku (0, 0, .. 0) in vektor denarja igralcev, ki je na začetku (t, t, .. t). Vektorja si shrani vsak igralec, da si lahko beleži potek igre.

6.2.2 Mešanje kart

Igralci zašifrirajo številke od 1 do 52, ki predstavljajo kupček kart. Število r, ki se uporabi za spremembo tajnopisa, je pri prvem šifriranju po ElGamalu v tem primeru javno, da vsi igralci vedo, da so res bile zašifrirane omenjene številke. Karte se nato preda prvemu igralcu, ki premeša zašifrirane številke s permutacijo π in spremeni šifrirana besedila po postopku iz podpoglavja 4.3 z naključnim r'. Zraven doda še dokaz o pravilnem mešanju ZKSH(π, r', c ,c') in poda karte naslednjemu igralcu, ki izvede enak postopek. To se ponavlja dokler kupčka ne premešajo vsi igralci. Na koncu dobimo kupček zašifriranih premešanih kart. Vsak igralec nato z deležem skritega ključa podpiše premešan kupček in podpisan kupček pošlje ostalim igralcem. Igralci nato preverijo podpise in jih shranijo kot dokaz o uspešnem mešanju kart.

6.2.3 Obvezni začetni stavi

Začnemo z majhno obvezno začetno stavo. Igralec P_{ssb}, ki ima obvezno majhno stavo pošlje podpisano sporočilo, ki signalizira, da stavi obvezno majhno stavo. Poleg tega pošlje podpisana vektorja za trenutno velikost stav igralcev in trenutno vrednost denarja, ki jo ima vsak igralec pri sebi po opravljeni začetni obvezni stavi. Če se ostali igralci strinjajo s pravilnostjo poteze, podpišejo sporočilo o uspešno opravljeni potezi skupaj s stanjem po opravljeni potezi. Obvezna velika začetna stava se izvede na enak način.

6.2.4 Vlečenje karte

Ko igralec vleče karto c_i, pošlje zahtevo po odšifriranju izbrane karte ostalim igralcem po privatnem kanalu. Vsak igralec P_j odšifrira karto s svojim deležem skritega ključa s_j in izračuna d_j. To in dokaz pravilnosti DLEQ(g, h_j, c_i, d_j) pošlje igralcu, ki je vlekel karto (odšifriranje iz podpoglavja 4.3). Vsak izmed igralcev na ta način izbere dve karti. Ko igralec vleče obe karti, sporoči ostalim igralcem, da je uspešno pridobil karte s podpisanim sporočilom akcije. Ta sporočila se nato shranijo kot dokaz o uspešni potezi vleke kart. Ostali igralci ne poznajo čistopisa vlečene karte, poznajo pa njen tajnopis.

6.2.5 Stave

Potek stav je opisan v poglavju 5. Ko je nek igralec P_i na vrsti, se odloči za eno izmed petih potez **FOLD**, **CALL**, **CHECK**, **RAISE**, **ALL IN** nato posodobi vektorja stav in trenutnih vrednosti denarja igralcev glede na svojo potezo. Potezo in nova vektorja stav podpiše s svojim podpisom in jih pošlje ostalim igralcem. Ostali igralci nato preverijo podpis ter če je bila sprememba vektorjev v skladu z akcijo in akcija v skladu s pravili pokra. Če to drži, igralci shranijo stanje stav in denarja igralcev. Nato podpišejo uspešen korak stav s spremenjenimi vektorji in podpisan del delijo z drugimi kot dokaz pravilne poteze stavljenja.

6.2.6 Javno odpiranje kart iz kupčka

V pokru se v poteku enega kroga igre odpre 5 kart, ki jih vidijo vsi igralci. Postopek se izvede tako, da vsi igralci izberejo isto karto s postopkom za vlečenje karte. Vsi igralci nato podpišejo sporočilo, da so izvlekli javno karto in vsebino te karte. Podpisana sporočila se shranijo za dokaz uspešnega odprtja javne karte.

6.2.7 Odkrivanje kart

Igralci javno pokažejo svoje karte ob koncu igre. Igralec na vrsti za kazanje kart lahko dovoli ostalim igralcem, da izvedejo operacijo za vlečenje karte na njegovih kartah ali pa pošlje vsem igralcem podpisano sporočilo, da bo karte odvrgel in se ne bo potegoval za zmago. Vsaka pravilno izvedena poteza je shranjena in podpisana s strani vseh igralcev.

6.2.8 Razporeditev zastavljenega denarja

Ko se igra konča, vsak igralec pri sebi izračuna novo stanje vektorja, ki vsebuje velikost premoženja igralcev in stave ponastavi na 0. Igralci nato to podpišejo in pošljejo ostalim igralcem. Če se vsi strinjajo, posodobijo vektorje in shranijo podpisana sporočila kot dokaz dogovora. Vsak igralec ima potem možnost, da se odloči, če bo igral naprej. Sporočilo z odločitvijo podpiše in pošlje ostalim. Če kateri od igralcev odide, se ga označi za neaktivnega. Novi javni ključ h je zmnožek deležev javnih ključev h_i od vseh aktivnih igralcev. Vsi nato izračunajo kateri igralci imajo obvezne začetne stave, vektorja trenutnih zneskov in velikosti stav ter kateri igralci so aktivni za naslednjo igro.

6.2.9 Odziv na obnovitev

Če kateri izmed igralcev poda zahtevo za obnovitev, morajo vsi ostali igralci deliti zadnjo shranjeno potezo s podpisi vseh igralcev ter svoje trenutno stanje. Če ima kateri izmed igralcev zastarelo zadnjo shranjeno potezo s podpisi ostalih igralcev, ta igralec pošlje zahtevo funkcionalnosti F, da obnovi izgubljene poteze.

7 Dokaz varnosti protokola

Če je odločitveni problem Diffie-Hellman težek in, če je shema podpisovanja EUF-CMA varna, potem je protokol varen v modelu *naključnega preroka* (angl. Random Oracle) v primeru pasivnih napadalcev.

Pri odločitvenem problemu Diffie-Hellman imamo ciklično grupo G z opreacijo množenja in generatorjem g. Če se izbere tri naključne elemente a,b in c iz grupe G se želi za trojico javnih ključev g^a, g^b, g^c ugotoviti ali $g^{ab} = g^c$. Zahteva je malo strožja od problema diskretnega logaritma, ker se ne želi, da bi se lahko iz g^a in g^b izračunalo g^{ab}. Primer družine takih grup Q_p se dobi tako, da se najde

nek $p = 2r + 1$, kjer sta p in r praštevili. Grupa Q_p vsebuje vse q za katere velja $q = x^2 (mod\ p) \mid x \in Z_p$ (kvadratni ostanki).

EUF-CMA *varnost pred ponaredbo pri napadu z izbranim sporočilom* (angl. Existential Unforgeability under chosen Message attack) poda naslednje zagotovilo. Igralec zgenerira javni in zasebni ključ. Napadalec ima dostop do javnega ključa. Napadalec nato zahteva podpisana sporočila, katerih čistopis izbere. Tako ima izbrane čistopise in za vsak čistopis tudi podpisano različico. EUF-CMA zagotavlja, da napadalec ne zmore zgenerirati novega para čistopisa in podpisane različice iz prejšnih parov. ElGamalov kriptosistem ima to lastnost.

Za dokaz varnosti z modelom *naključnega preroka* se uporabi naključnostni simulator S, katerega operacije so izvedene v polinomskem času. Simulator preroka S simulira poteze poštenih igralcev H (honest players) in funkcionalnosti F. Igra pa proti nepoštenimi igralci C (corrupted players). Prerok ima dostop do simulatorjev dokazov brez odkritja znanja. To mu poda možnost, da lahko v procesu mešanja kupčka sledi pozicijam kart. Poleg tega lahko ustvari dokaz z uporabo napačnih deležev odšifrirane vrednosti, ki pa se na koncu še zmeraj prevedejo v pravilno odšifrirano vrednost, ker je S pošten. Te dve lastnosti mu zagotovita varnost pred goljufijami. Radi bi dokazali, da je možnost goljufije enako verjetna kot pri normalni igri. Dokaz ZKSH iz podpoglavja 4.4 nam zagotovi pravilnost mešanja, ne da bi vedeli pozicije kart. Ker je skrivni delež ključa igralca P_i izbran naključno, so tudi vrednosti uporabljene v dokazu pravilnosti deleža odšifrirane vrednosti naključne. V primeru preroka S in poštenega igralca so s strani nasprotnika vrednosti za dokaz pravilnega deleža odšifrirane vrednosti naključne. Igra s prerokom S je torej enako varna kot navadna igra.

8 Zaključek

V nalogi je predstavljen protokol Kaleidoscope, ki je prvi decentraliziran protokol za igranje miselnega pokra, ki je hkrati hiter in ima formalen dokaz za varnost protokola. Dokaz varnosti temelji na tem, da se uporabi znane metode za katere poznamo varnostna zagotovila. Hkrati pa omeji delo na *veriženih skupkih* in s tem pridobi na hitrosti. Na koncu tega članka, ki opisuje protokol, ostajata dva vprašanja. Ali je protokol varen, če izvajamo več instanc protokola vzporedno in, ali bi bil protokol še zmeraj varen, če bi spremenili vrstni red operacij s kartami, da bi igrali kako drugo igro s kartami (npr. Blackjack). Zato menim, da so trenutno protokoli s centrom zaupanja še zmeraj bolj varna izbira. Verjamem pa, da se bo to v parih letih spremenilo, ko bojo izšli novi članki, ki bojo razrešili ti dve odprti vprašanji.

References

Adam Barnett and Nigel P Smart. 2003. Mental poker revisited. In *IMA International Conference on Cryptography and Coding*, pages 370–383. Springer.

Stephanie Bayer and Jens Groth. 2012. Efficient zero-knowledge argument for correctness of a shuffle. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 263–280. Springer.

Ranjit Bentov, Iddo ij Kumaresan and Andrew Miller. 2017. Instantaneous decentralized poker. In *International conference on the theory and application of cryptology and information security*, pages 410–440. Springer.

Jordi Castella-Roca, Josep Domingo-Ferrer, Andreu Riera, and Joan Borrell. 2003. Practical mental poker without a ttp based on homomorphic encryption. In *International Conference on Cryptology in India*, pages 280–294. Springer.

Jordi Castella-Roca and Josep Sebé, Francesc ij Domingo-Ferrer. 2005. Dropout-tolerant ttp-free mental poker. In *International Conference on Trust, Privacy and Security in Digital Business*, pages 30–40. Springer.

Don Coppersmith. 1985. Cheating at mental poker. In *Conference on the Theory and Application of Cryptographic Techniques*, pages 104–107. Springer.

Bernardo David, Rafael Dowsley, and Mario Larangeira. 2018. Kaleidoscope: An efficient poker protocol with payment distribution and penalty enforcement. In *International Conference on Financial Cryptography and Data Security*, pages 500–519. Springer.

Ranjit Kumaresan, Tal Moran, and Iddo Bentov. 2015. How to use bitcoin to play decentralized poker. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 195–206.

R Lipton. 1981. How to cheat at mental poker. *Proc. AMS Short Course on Cryptography, Jan. 1981.*

Adi Shamir, Ronald L Rivest, and Leonard M Adleman. 1981. Mental poker. In *The mathematical gardner*, pages 37–43. Springer.