Decentralizirane biometrijske metode autentifikacije s očuvanjem privatnosti

Gardijan, Andrej

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET
Sveučilišni diplomski studij računarstva

Diplomski rad

Decentralizirane biometrijske metode autentifikacije s očuvanjem privatnosti

Rijeka, ožujak 2023.

Andrej Gardijan
0069082744

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET
Sveučilišni diplomski studij računarstva

Diplomski rad

Decentralizirane biometrijske metode autentifikacije s očuvanjem privatnosti

Mentor: Prof. dr. sc. Kristijan Lenac

Rijeka, ožujak 2023.

Andrej Gardijan
0069082744

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET
POVJERENSTVO ZA DIPLOMSKE ISPITE

Rijeka, 21. ožujka 2022.

Zavod: **Zavod za računarstvo**
Predmet: **Napredni operacijski sustavi**
Polje: **2.09 Računarstvo**

ZADATAK ZA DIPLOMSKI RAD

Pristupnik: **Andrej Gardijan (0069082744)**
Studij: Sveučilišni diplomski studij računarstva
Modul: Računalni sustavi

Zadatak: **Decentralizirane biometrijske metode autentifikacije s očuvanjem privatnosti / Privacy-preserving decentralized biometric authentication methods**

Opis zadatka:

Istražiti postojeće tehnike za očuvanje privatnosti biometrijskih metoda koje se ne oslanjaju na centralni modul za autentifikaciju. Predložiti i demonstrirati koncept rješenja za privatnu decentraliziranu biometrijsku autentifikaciju.

Rad mora biti napisan prema Uputama za pisanje diplomskih / završnih radova koje su objavljene na mrežnim stranicama studija.

Zadatak uručen pristupniku: 21. ožujka 2022.

Mentor:

Prof. dr. sc. Kristijan Lenac

Predsjednik povjerenstva za diplomski ispit:

Prof. dr. sc. Kristijan Lenac

Izjava o samostalnoj izradi rada

Izjavljujem da sam samostalno izradio ovaj rad.

Rijeka, ožujak 2023.

_____
Andrej Gardijan

Zahvala

Zahvaljujem mentoru na podršci tijekom pisanja ovoga rada i korisnim raspravama i savjetima. Zahvaljujem svojim kolegama, prijateljima i ponajviše obitelji na podršci tijekom studiranja.

Sadržaj

Popis slika

Popis tablica

Poglavlje 1

Uvod

Biometriju možemo definirati kao istraživanje mogućnosti prepoznavanja osoba na temelju njihovih fizičkih i/ili bihevioralnih (psiholoških) karakteristika. Biometrijska autentifikacija je najraširenija primjena biometrijskog prepoznavanja. Odnosi se na proces utvrđivanja ili potvrđivanja identiteta kao autentičnog, odnosno ukoliko je onaj tko je vlasnik snimljenog biometrijskog obilježja onaj za kog se predstavlja [1].

Decentralizacija je proces disperzije moći od centralnih autoriteta. Povećanjem stupnja decentralizacije biometrijske autentifikacije moguće je naslijediti prednosti podložne tehnologije, odnosno blockchaina kao infrastrukture. Blockchain u prvotnoj namjeni predstavlja decentraliziranu infrastrukturu. Decentralizacija je ključno svojstvo blockchaina koje omogućuje sigurnost sustava te obećava distribuiranu *peer-to-peer* mrežu [2] [3] [4]. Sustav kojem je podložna infrastrukura bazirana na verifikaciji, decentralizaciji itd. ima veću robusnost od centraliziranog sustava baziranog na povjerenju. Međutim, stvarna razina decentralizacije uvelike varira u svijetu blockchaina [5] [6].

Poželjne značajke koje blockchain pruža biometriji su nepromjenjivost, dostupnost i univerzalni pristup. Ova svojstva se u biometrijskim aplikacijama koriste za osiguranje biometrijskih predložaka i osiguravanje privatnosti u biometrijskim sustavima." U tom tekstu i dalje nije jasno kako se osigurava privatnost.

Prema izjavi Europskog nadzornika za zaštitu podataka, gdje se navodi: “Središnja baza podataka za razliku od decentralizirane baze podataka implicitno povećava

rizike od zlouporabe i lakše potiče želju za korištenjem sustava izvan svrhe za koju je izvorno namijenjen" [7].

Nepromjenjivost se često povezuje sa sigurnošću pohrane biometrijskih predložaka i čak se identificira kao glavna svrha kombinacije istih s blockchainom. Međutim, takva vrsta nepromjenjivosti može ugroziti privatnost pojedinaca, posebno kada se obrađuju njihovi biometrijski podaci. Biometrijske osobine pojedinca su nepromjenjive, te se stoga pohrani i obradi biometrijskih predložaka dobivenih na temelju tih osobina treba pristupiti s pažnjom.

Na samom početku rada biti će definirani osnovni pojmovi potrebni za razumijevanje rada, odnosno; što je to biometrija, *blockchain* mreža; kako se ona dijeli te koje su njene komponente. Potom će u istom poglavlju biti riječi o interplanetarnom datotečnom sustavu koji omogućuje izradu nezamjenjivih tokena, nakon čega će i oni biti spomenuti. Naposljetku, definiraju se osnovni pojmovi i postupci u biometriji. Nakon toga će biti govora o razmatranim rješenjima i potencijalno korisnim projektima. Poslije toga predloženo je rješenje inspirirano postojećim obrađenim rješenjima, a nadograđeno metodama opisanim u ovome radu. Rješenje kombinira IPFS za pohranu podataka te se kombinacijom s blockchainom dodaje potvrda integriteta i programabilnost. Objašnjeni su i usko vezani pojmovi kao što je *zkEVM*, MPC, kvantna sigurnost te homomorfna enkripcija. Na kraju poglavlja, spominju se razmatrane blockchain mreže te osnovni pristupi pohrane podataka, bilo cjelovitih podataka ili njihovih sažetaka s prednosti očuvanja integriteta. Pred kraju nalazi se implementacija nezamjenjivog tokena na Cardano blockchainu U tom poglavlju opisano je okruženje, potom postavljanje IPFS čvora, te instalacija Nix upravitelja paketa, kako bi se mogle izgraditi sve izvršne datoteke za Cardano mrežu iz izvornog koda. Tada su podaci preneseni na IPFS, te je detaljno opisan postupak stvaranja nezamjenjivog tokena koristeći lokalni Cardano čvor. Na kraju poglavlja nalazi se primjer stvorenog nezamjenjivog tokena s njegovim karakteristikama na preview Cardano testnoj mreži. Završni dio rada uključuje diskusiju navedenog kao i zaključak.

Poglavlje 2

Biometrija

U široj perspektivi biometrija je statističko istraživanje o biološkim fenomenima; to je korištenje matematike i statistike u razumijevanju živih bića [8].

U užoj perspektivi biometriju možemo definirati kao istraživanje mogućnosti prepoznavanja osoba na temelju njihovih fizičkih i/ili bihevioralnih (psiholoških) karakteristika [9].

Za razliku od konvencionalnih metoda autentifikacije temeljenih na znanju (npr. lozinka) ili posjedovanju (npr. pametna kartica), biometrijsko prepoznavanje oslanja se na koncept inherentnosti, tj. tko je netko, što ga čini otpornijim protiv lažnih aktivnosti kao što su krivotvorenje, lažiranje, itd., u usporedbi s prethodno navedenim metodama [1].

Biometrijska karakteristika je fizička karakteristika ili karakteristika ponašanja pojedinca koja se može koristiti za prepoznavanje istog. U užoj perspektivi biometrije fizičke karakteristike su genetski implicirane (moguće pod utjecajem okoliša) karakteristike (poput nečijeg lica, šarenice, mrežnice, prsta, vaskularne strukture itd.). Bihevioralne ili psihološke karakteristike su karakteristike koje osoba stječe ili nauči tijekom života (kao što je vlastoručni potpis, hod osobe, dinamika tipkanja ili karakteristike glasa). Ove se definicije gotovo lako prevode u širu perspektivu biometrije. Ovisno o broju karakteristika koje se koriste za prepoznavanje biometrijski sustavi mogu biti unimodalni (kada se koristi samo jedna biometrijska karakteristika) ili multimodalni (ako se koristi više od jedne karakteristike).

Biometrijska struktura je posebna značajka neke biometrijske karakteristike koja se može koristiti za prepoznavanje (na primjer, biometrijska struktura karakteristična za prst je struktura papilarnih linija i minucija, za hod karakteristična je struktura pokreta tijela tijekom ljudskog hoda itd.).

Biometrijski uzorak predstavlja neku od navedenih izmjerenih veličina pojedinca.

Biometrijski predložak ili ekstrahirana struktura je količina ili skup količina dobivenih svjesnom primjenom ekstrakcije biometrijskih značajki ili metode predprocesiranja na biometrijskom uzorku. Ti su predlošci obično pohranjeni u biometrijskoj bazi podataka i koriste se kao referenca tijekom prepoznavanja ili upisa u biometrijski sustav.

2.1 Biometrijski procesi

Biometrijski sustavi oslanjaju se na nekoliko diskretnih procesa: upis *(eng. enrollment)*, snimanje uživo *(eng. live capture)*, izdvajanje predložaka *(eng. template extraction)* i usporedbu predložaka *(eng. template comparison)*.

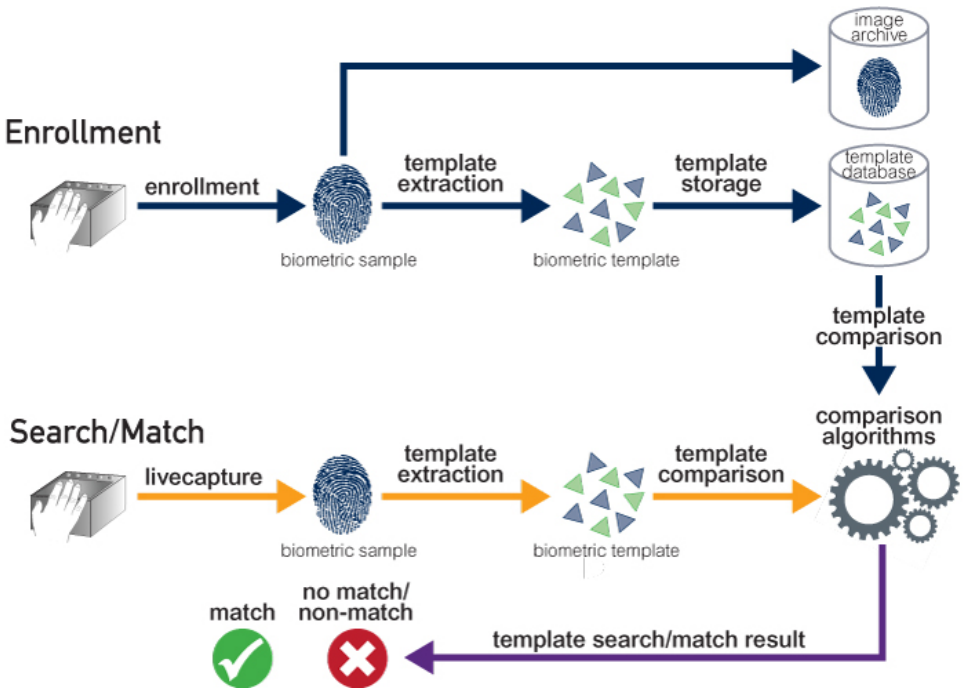
Svrha upisa je prikupljanje i arhiviranje biometrijskih uzoraka te generiranje numeričkih predložaka za buduće usporedbe. Arhiviranjem neobrađenih uzoraka mogu se generirati novi zamjenski predlošci u slučaju da se novi ili ažurirani algoritam usporedbe uvede u sustav. Prakse koje olakšavaju upis visokokvalitetnih uzoraka ključne su za dosljednost uzorka i poboljšavaju ukupnu izvedbu podudaranja, što je osobito važno za biometrijsku identifikaciju pretraživanjem "jedan prema više" [10].

Možemo razlikovati "snimanje uživo" od upisa kao procesa prikupljanja biometrijskih uzoraka uživo nakon pristupa ili pokušaja identifikacije i njihove usporedbe s "galerijom" prethodno upisanih predložaka.

Ekstrakcija predloška zahtijeva obradu signala neobrađenih biometrijskih uzoraka (npr. slika ili audio uzoraka) kako bi se dobio numerički predložak. Predlošci se obično generiraju i pohranjuju nakon upisa kako bi se uštedjelo vrijeme obrade pri budućim usporedbama. Usporedba dvaju biometrijskih predložaka primjenjuje algoritamske proračune za procjenu njihove sličnosti. Nakon usporedbe, dodjeljuje se rezultat podudaranja. Ako je iznad određenog praga, predlošci se smatraju odgo-

varajućim.

Obično su algoritmi za izdvajanje i usporedbu biometrijskih predložaka vlasnički *(eng. proprietary)* (različiti i tajni) i stoga se ne mogu koristiti s onima različitih dobavljača u istom sustavu (npr. za usporedbu predložaka koje su generirali različiti proizvodi ili korištenje odgovarajućeg algoritma jedne tvrtke za usporediti predloške generirane algoritmima drugog). Postoje i iznimke napravljene da budu interoperabilne za provjeru jedan-na-jedan te su stoga idealne za kompaktnu pohranu na pametnim karticama ili putnim dokumentima [11].



Slika 2.1 Biometrijski proces [10]

2.2 Biometrijska autentifikacija

Biometrijska autentifikacija odnosi se na sigurnosni postupak koji uključuje korištenje jedinstvenih bioloških karakteristika pojedinaca kao što su mrežnice, šarenice, glasovi, karakteristike lica i otisci prstiju s ciljem potvrde da je pojedinac onaj za

kojeg se predstavlja da je. Ovaj se proces koristi za kontrolu pristupa fizičkim i digitalnim resursima, kao što su zgrade, sobe i različiti uređaji [12] [13]. Primjer jednog takvog sustava može biti računalo koje će skenirati osobu u potrazi za inherentnim atributima - na primjer, predloškom za prepoznavanje lica, a zatim će usporediti predložak dobiven od karakteristika pojedinca s predloškom pohranjenim u bazi podataka. Ako predložak skeniranog uzorka nekog atributa odgovaraju predlošku, osobi je dopušten ulazak u sustav.

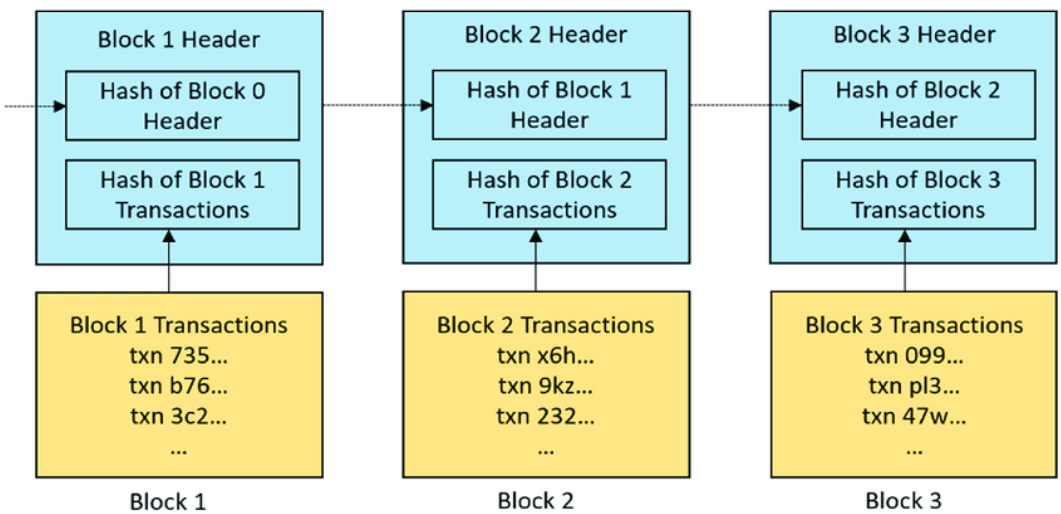
Slika 2.2 Metode za biometrijsku autentifikaciju [12]

Poglavlje 3

Blockchain

Blockchain predstavlja računalnu tehnologiju za pohranu i obradu transakcija koja je sigurna (bez mogućnosti gubitka ili promjene podataka), transparentna (jednostavna provjera i praćenje) i bez povjerenja (povjerenje u transakcije bez ikakvog posrednika).

Sastoji se od računala povezanih u mrežu, svaki od kojih se naziva čvor te sadrži lokalnu kopiju lanca. Glavna ideja *blockchaina* jest "Verificiraj, ne vjeruj.[14]

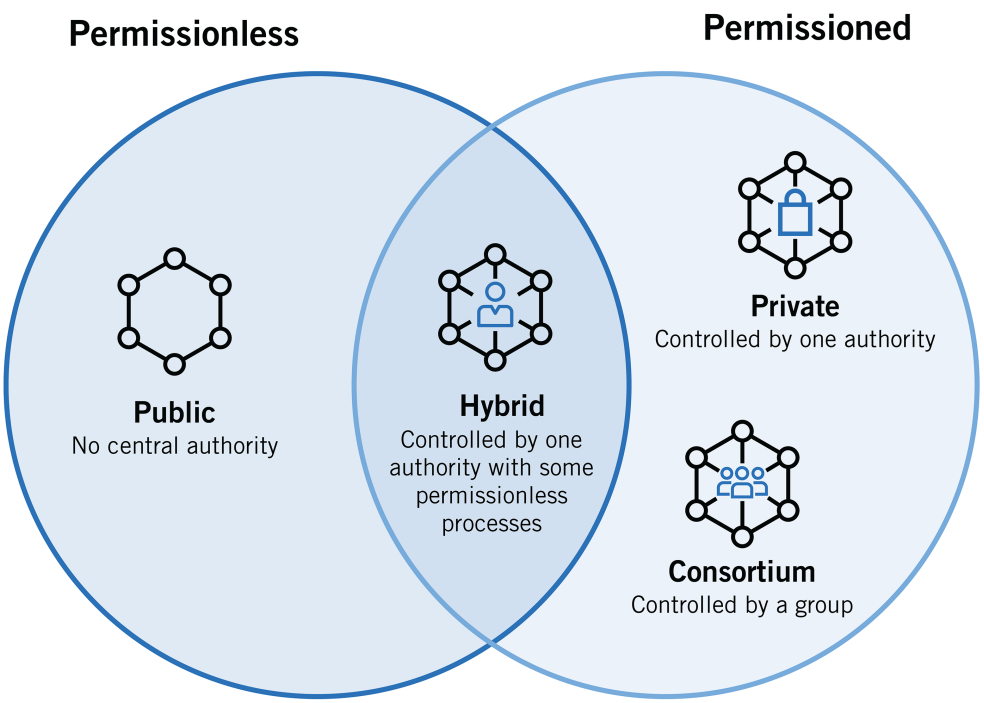


Slika 3.1 Pojednostavljeni prikaz formiranja blokova kako bi se oformio *blockchain* [14]

3.1 Podjela po načinu upravljanja i pristupu

Blockchain mreže mogu se kategorizirati na temelju njihovog modela dopuštenja, koji određuje tko ih može održavati (odnosno objavljivati blokove). Ako bilo tko može objaviti novi blok, govorimo o permissionless modelu.

Ako samo određeni korisnici mogu objavljivati blokove, radi se o permissioned modelu. Jednostavno rečeno, permissioned blockchain mreža je poput korporativnog intraneta koji je kontroliran, dok je permissionless blockchain mreža poput javnog interneta, gdje svatko može sudjelovati. Permissioned blockchain mreže obično se postavljaju za grupu organizacija i pojedinaca koji se tipično nazivaju konzorcij. Ovu razliku potrebno je razumjeti jer utječe na neke od komponenata blockchaina [15].

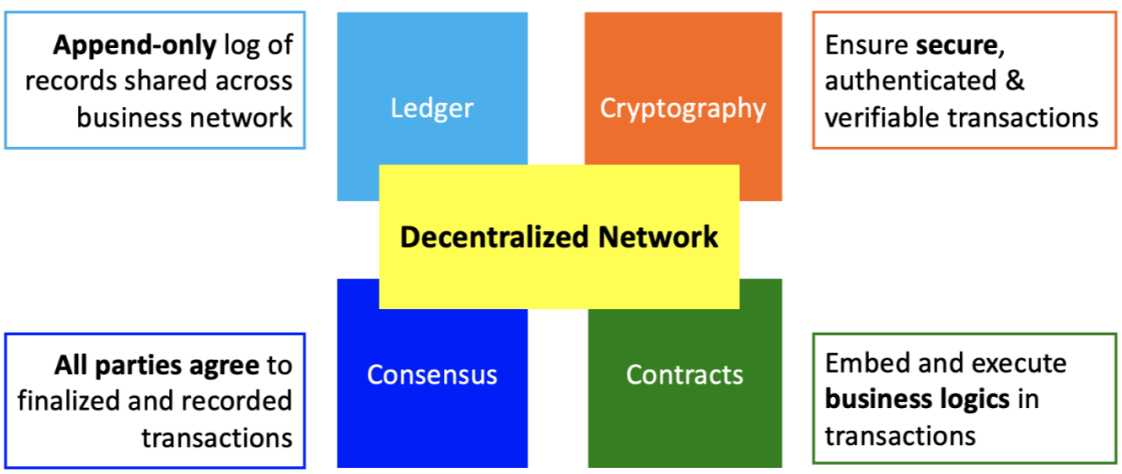


Slika 3.2 Permissioned i Permissionless strukture blockchaina [16]

3.2 Komponente Blockchain mreže

Tehnički gledano, najpotpunija definicija blockchaina trebala bi ga vidjeti kao decentralizirani računalni sustav od pet sastavnih komponenti 3.3:

1. decentralizirano umrežavanje
2. kriptografija
3. distribuirani konsenzus
4. transakcijska knjiga
5. pametni ugovori



Slika 3.3 Pet sastavnih komponenti blockchaina

3.2.1 Decentralizirano umrežavanje

Blockchain se oslanja na decentraliziranu mrežu računala, zvanih blockchain čvorovi, koji pridonose računalne resurse za pomoć u pohrani i obradi transakcija. Ova računala rade autonomno i međusobno komuniciraju na način peer-to-peer (P2P).

Većina blockchain mreža uključujući Bitcoin usvajaju nestrukturiranu P2P topologiju; tj. čvor proizvoljno bira svoje susjede.

Neke mreže kao što je Ethereum koriste strukturiranu topologiju kao što je Kademlia Distributed Hash Table [17] za optimizaciju P2P komunikacija. Nestrukturirani P2P može biti manje učinkovit od strukturiranog P2P-a.

Ethereum koristi Kademlia, ali samo kao pomoćni dodatak [18]; drugim riječima, radi i dalje s bilo kojom nestrukturiranom P2P topologijom, iako je tada samo manje učinkovit.

3.2.2 Kriptografija

Kriptografske metode koje se koriste u blockchainu pružaju matematički dokaz da blockchain mora funkcionirati kako treba. Kriptografski hash koristi se za povezivanje blokova podataka u lanac tako da nema izmjene podataka nakon zapisa u blockchain.

Svaka transakcija šifrirana je s javnim ključem pošiljatelja, odnosno koristi se kriptografija bazirana na javnom ključu.

Odabir kriptografije za korištenje određuje izvedbu i jamstva blockchaina. Na primjer, Dogecoin [19] blockchain klonira Bitcoin, ali koristeći jednostavnije kriptografske funkcije za povećanje propusnosti broja transakcija; rudarenje u Dogecoinu temelji se na SCRYPT-u koji je brži i lakši za izračun od secure hashing algorithm 256 (SHA256) koji se koristi u Bitcoinu [20]. To, međutim, rezultira slabijom sigurnošću, manjom robusnosti od napada nepoštenih čvorova.

3.2.3 Transakcijska knjiga

Kao tehnologiju pohrane, Blockchain koristi digitalnu knjiga koja ima svojstvo pohrane transakcija kronološki u blokovima; odnosno blokovi se uvijek nadodaju na prethodni lanac blokova. Ovo je zadana struktura podataka transakcijske knjige za gotovo sve blockchain mreže.

Međutim, neke blockchain mreže, na primjer, Hedera [21] i Fantom [22], dizajniraju transakcijsku knjigu kao directed acyclic graph (DAG) blokova (ili transakcija) umjesto lančane strukture koja može samo dodavati blokove.

Prva metoda nudi jednostavnost, dok je druga učinkovitija u obradi transakcija (primjerice, traženje istih je brže).

Struktura transakcijske knjige, struktura bloka i broj transakcija u bloku su važni parametri koje je potrebno uzeti u obzir pri dizajniranju transakcijske knjige pojedinog Blockchain-a.

3.2.4 Distribuirani konsenzus

Kada treba donijeti odluku, na primjer hoće li transakcija biti važeća, nema središnjeg tijela koje odlučuje. Umjesto toga, do odluke se dolazi na temelju konsenzusa postignutog među sudjelujućim čvorovima. Stoga, blockchain mreža mora imati konseznusni protokol kako bi se osiguralo da je svaka transakcija ili blok dodan u blockchain jedna i jedina verzija istine oko koje se slažu svi čvorovi.

Proof-of-Work konsenzus [20], pridodaje veću moć odlučivanja čvorovima s većom računalnom snagom, te je usvojen u ranim blockchain mrežama (Bitcoin, Litecoin, Ethereum - izvorna verzija).

Proof-of-Stake konsenzus [23], daje veću moć odlučivanja čvorovima s većim financijskim ulogom, popularan je među današnjim blockchain mrežama; prva funkcionirajuća upotreba za kriptovalute bila je u Peercoinu 2012. [24].

Izbor konsenzus protokola je najkritičnije razmatranje u dizajniranju blockchain mreže.

3.2.5 Pametni ugovori

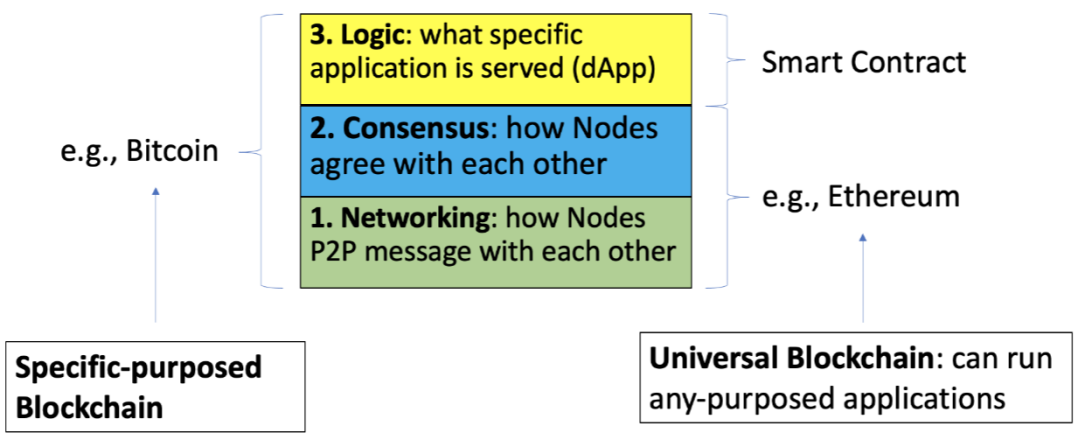
Blockchain se može smatrati nekonvencionalnom vrstom računala za obavljanje određenih zadataka. Umjesto da bude računalo koje integrira računalne procesorske jedinice, blockchain je decentralizirano računalo koje koristi stotine ili tisuće računala bilo gdje u svijetu.

Aplikacije koje rade na blockchainu implementiraju se kao "pametni ugovori", izraz koji je skovao Nick Szabo 1990-ih [25].

Pametni ugovor nije ništa drugo nego računalni program; izraz se koristi jer aplikacija postavljena na blockchain uvijek funkcionira ispravno kao što je programirano, poput izvršavanja uvjeta za pravni ugovor. Ovakav je ugovor pametan jer se automatski izvršava bez ljudske intervencije.

3.3 Blockchain kao računalo

Blockchain možemo promatrati kao računalo čija se arhitektura sastoji od tri sloja, vidljivo na slici 3.4, P2P, mrežni sloj, konsenzusni sloj i sloj logike.



Slika 3.4 Arhitektura Blockchaina kao računala

Na primjer, Bitcoin je blockchain računalo koje implementira sve ove slojeve, dok

Ethereum implementira prva dva sloja, ostavljajući logički sloj za primjenu softverskim developerima. Bitcoin je blockchain računalo za specifičnu svrhu koje obavlja samo jednu aplikaciju: stvoriti digitalnu valutu, kriptovalutu, sa funkcijom prijenosa između korisnika.

Ovo je prirodna evolucija u računarstvu. Računarstvo u oblaku je zamijenilo stolno računarstvo zbog smanjenja troškova održavanja IT sustava za tvrtke i u isto vrijeme za učinkovitije korištenje računalnih resursa.

To je sve na jednom mjestu kako bi se zadovoljile sve računalne potrebe da bi tvrtke mogle više vremena usredotočiti na svoju poslovnu logiku.

U usporedbi s računarstvom u oblaku, Blockchain nudi benefit decentralizacije i jamstva bez potrebe za povjerenja. Pružatelj usluga u oblaku ima moć manipulirati računalom u oblaku; moramo vjerovati takvoj organizaciji. Blockchain je bez povjerenja i svatko može postati učesnik.

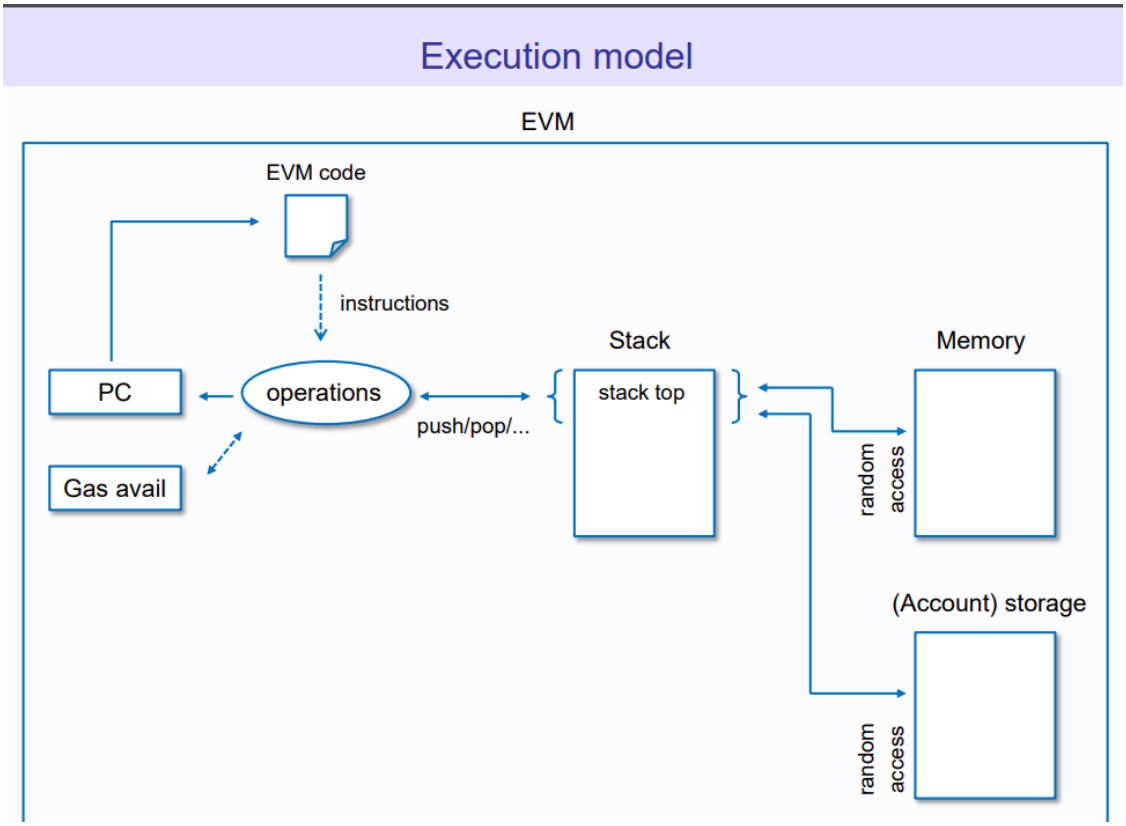
Način rada EVM

EVM (*Ethereum Virtual machine*) je stroj stanja koji se pomiče iz starog stanja u novo stanje kao odgovor na neke ulaze. Svako izvršenje pametnog ugovora pokreće promjenu stanja EVM-a (naziva se "prijelaz stanja").

Tijekom transakcije pametnog ugovora događa se sljedeće [26]:

1. Bajtni kod pametnog ugovora (kompajliran iz izvornog koda) učitava se iz EVM-ove pohrane i izvršavaju ga peer-to-peer čvorovi na EVM-u. Čvorovi koriste iste transakcijske ulaze, što jamči da svaki čvor dolazi do istog rezultata (inače ne mogu postići konsenzus)
2. EVM op kodovi (sadržani u bajt kodu) komuniciraju s različitim dijelovima EVM (memorija, pohrana i stog). Operativni kodovi izvode operacije čitanja i pisanja—čitanje (dobivanje) vrijednosti iz pohrane stanja i pisanje (slanje) novih vrijednosti u pohranu EVM-a.
3. EVM operacijski kodovi izvode izračunavanje vrijednosti dobivenih iz pohrane stanja prije vraćanja novih vrijednosti. Ovo ažuriranje rezultira prelaskom EVM-a u novo stanje (transakcije se iz tog razloga nazivaju "prijelazi stanja").

Ovo novo stanje repliciraju drugi čvorovi i ostaje dok se ne izvrši druga transakcija.



Slika 3.5 Dijagram izvršavanja programa unutar Ethereum virtualnog stroja [27]

3.3.1 Layer 2 Blockchain mreže

Kritični aspekt blockchaina tiče se njegove skalabilnosti. U trenutku pisanja, skalabilnost se smatra uskim grlom blockchain infrastrukture [28]. Blockchain bi potencijalno mogao konkurirati najvećim poslužitzeljima elektroničkog plaćanja. Međutim, ograničen je time što može obraditi nekoliko transakcija u sekundi TPS.

Da bismo razjasnili problematiku, dovoljno je navesti dva najpoznatija blockchaina kao primjer: Bitcoin obrađuje 4,6 TPS, a Ethereum obrađuje oko 14,3 TPS (promjenjive vrijednosti), dok jedan od najvećih krugova elektroničkog plaćanja, Visa,

obrađuje oko 1736 TPS (i uspio je doseći vrhunce od 47000 TPS). Trenutno, Bitcoin blockchain generira novi blok svakih 10 minuta (vrijeme Generiranje bloka, TB) i veličina bloka B u lancu je 1 MB (1.048.576 bajtova). Prosječna veličina transakcije je 380 bajtova. Prema tome, broj transakcija koji se uklapa u Bitcoin blok TPB je:

$$\text{TPB} = \frac{\text{Block Size}}{\text{Average Transaction Size}} = \frac{1,048,576 \text{ Bytes}}{380 \text{ Bytes}} \approx 2,759 \text{ transactions} \tag{3.1}$$

Kao posljedica TB i TPB, broj transakcija u sekundi iznosi:

$$TPS = \frac{TPB}{TB} \approx \frac{2,759 \text{ transactions}}{600 \text{ seconds}} \approx 4.6 \text{ tps} \tag{3.2}$$

Suprotno onome što bi se moglo pomisliti, problem skalabilnosti ne može se riješiti jednostavnim modeliranjem njegovih parametara: B i TB. Zapravo, s obzirom na modeliranje parametara, potrebno je napraviti važno pojašnjenje: prilikom kreiranja novog bloka u blockchainu, ključni čimbenik koji treba uzeti u obzir je vrijeme releja TR potrebno za emitiranje novog bloka svakom čvoru na mreži.

Dakle, ova činjenica nameće donju granicu TB ispod koje se ne može ići. Ovaj TR prag omogućuje stalno ažuriranje svih čvorova u mreži. Dodatno, javlja se još jedan problem vezan uz TR prilikom povećanja veličine bloka B. Posljedično, mora se emitirati povećana količina informacija na mrežu.

Razmatramo primjer iz Bitcoin blockchaina radi jednostavni: u siječnju 2021. procijenjeno je oko 10 000 čvorova u Bitcoin mreži. Prosječno vrijeme za širenje bloka do 99% mreže je otprilike 14 sekundi. Stoga TB ne može pasti ispod praga od 14 sekundi. Inače bi se novi blok generirao prije nego bi većina čvorova u mreži primila stari blok. Problem vezan uz veličinu bloka postaje evidentan prilikom njegovog povećanja; 14 sekundi sekundi kao TR više nije dovoljno.

Najčešće korišten pristup za postizanje skalabilnog blockchaina općenito je poznat kao "Layer 2": osnovna ideja je izgradnja sloja koji upravlja transakcijama izvan blockchaina (ne u glavnom blockchainu i, u određenom smislu, neovisno od njega), čime se smanjuje opterećenje samog blockchaina i postiže veća transakcijska propusnost.

Naravno, kao što je opisano, kretanjem u smjeru skalabilnosti, posebice u smjeru transakcija izvan glavnog blockchaina, dovodi do problema u pogledu sigurnosti i decentralizacije, koje treba riješiti posebnim protumjerama.

Layer 2 rješenje je sekundarni protokol izgrađen na postojećem blockchainu; ideja koja stoji iza takvog rješenja može biti različite prirode, ali ključni koncept jest ta usluga obrade grupa transakcija i izvješćivanje samo o njihovom "sažetku" na glavnom blokchainu.

Blockchain *rollups* kompajliraju hrpu transakcija i pretvaraju ih u jedan jedini podatak i šalju ga glavnoj. Oni preuzimaju transakcije iz glavne mreže i obrađuju ih izvan lanca, pretvaraju ih u jedan podatak i šalju ih natrag u glavnu mrežu. Zbog toga se rollupovi također nazivaju "rješenjima za skaliranje izvan blockchaina".

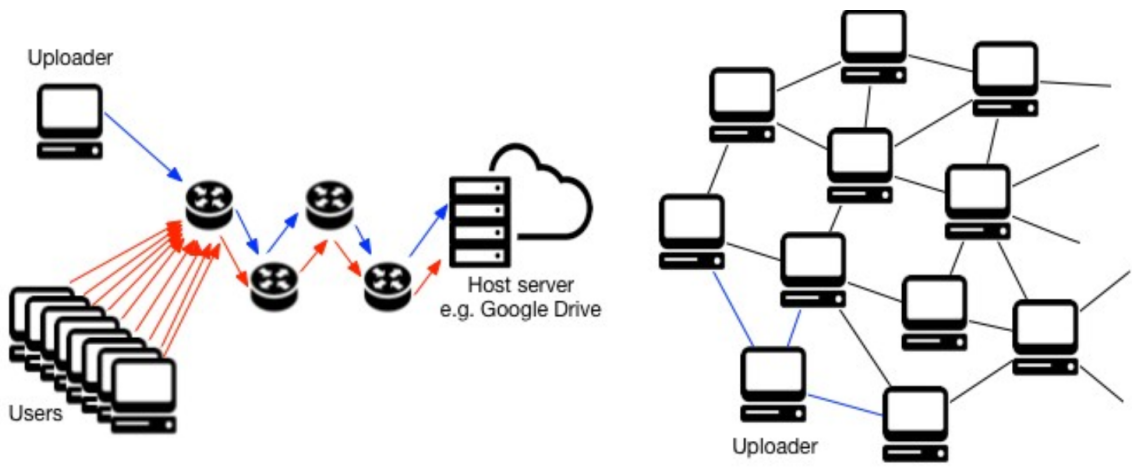
3.4 Interplanetarni datotečni sustav

Interplanetarni datotečni sustav *(eng. interplanetary filesystem (IPFS))* je distribuirani datotečni sustav koji spaja uspješne ideje iz prethodnih peer-to-peer sustava, uključujući DHT-ove, BitTorrent, Git i SFS [29].

Doprinos IPFS-a je pojednostavljenje, razvoj i povezivanje dokazanih tehnika u jedan kohezivni sustav, veći od zbroja svojih dijelova.

IPFS predstavlja novu platformu za distribuciju i verzioniranje podataka. IPFS je P2P; nijedan čvor nije privilegiran. IPFS čvorovi pohranjuju IPFS objekte lokalno. Čvorovi se povezuju sa svakim ostalim i prenose objekte. Ovi objekti predstavljaju datoteke i druge strukture podataka. IPFS protokol je podijeljen na podprotokole od kojih je svaki odgovoran za različite funkcije.

Drugim riječima, IPFS pruža visokopropusni model blokovske pohrane s hipervezama adresiranog sadržaja. To formira generalizirani Merkle DAG, podatkovnu struktura na kojoj se mogu graditi verzionirani datotečni sustavi, blockchainovi, pa čak i Permanent Web. IPFS kombinira distribuiranu hash tablicu, incentiviziranu razmjenu blokova, itd. IPFS nema jednu točku kvara, a čvorovi ne moraju vjerovati jedni drugima.

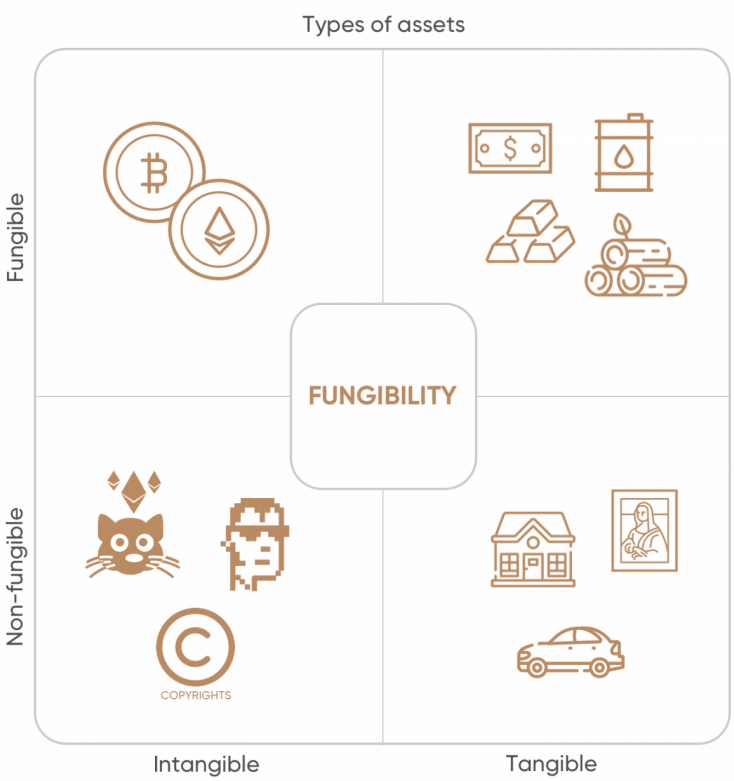


Slika 3.6 Tok podataka u centraliziranom sustavu i IPFS-u [30]

3.5 Nezamjenjivi tokeni

Nezamjenjivi token NFT jedinstvena je jedinica podataka na blockchainu koja se može povezati s digitalnim i fizičkim objektima kako bi se pružio nepromjenjivi dokaz vlasništva [31].

Podaci koje NFT sadrži mogu se povezati s digitalnim slikama, pjesmama, videozapisima, avatarima itd. Međutim, oni se također mogu koristiti kako bi se vlasniku NFT-a omogućio pristup ekskluzivnoj robi, ulaznicama za događaje uživo ili digitalnim događajima, ili se mogu povezati s fizičkom imovinom kao što su automobili, jahte i još mnogo toga.U tom smislu, NFT-evi omogućuju pojedincima da stvaraju, kupuju i prodaju artikle na lako provjerljiv način koristeći blockchain tehnologiju.



Slika 3.7 Usporedba zamjenjivog i nezamjenjivog svojstva [32]

Kriptovalute se mogu kupiti ili pretvoriti u fiat valute (dolare, eure, jene itd.) putem kripto mjenjačnica. Nasuprot tome, NFT je jedinstvena i nezamjenjiva imovina koja se kupuje pomoću kriptovalute. Može dobiti ili izgubiti vrijednost neovisno o valuti korištenoj za kupnju, baš kao popularna kartica za razmjenu ili unikatno umjetničko djelo. Stoga, NFT-evi su nezamjenjivi, a kriptovalute su zamjenjive.

Poglavlje 4

Pregled postojećih rješenja

Prevladavajući nedostatak pristupa većine postojećih rješenja za biometrijske sustave je da je još uvijek potrebno osigurati dostupnost podataka pohranjenih izvan *blockchaina*. Ako bi se ti podaci izgubili ili neovlašteno mijenjali, čak i kada bi se ta izmjena uvijek primijetila, održivost sustava bila bi ugrožena. Dodatna mana tog pristupa u aspektu privatnosti je što su podaci pohranjeni unutar centraliziranog sustava, s obzirom da se pohranom na *blockchain* štiti samo njihov integritet.

4.1 Blockchain i biometrija

4.1.1 Euklidska udaljenost

U radu [33] koristi se Euklidska udaljenost za autentifikaciju na način da se uspoređuju euklidske udaljenosti između parova vektora biometrijskih predložaka. Da bi se prevladao nedostatak izvorne podrške za aritmetiku s pomičnim zarezom, decimalni brojevi se predstavljaju kao cijeli brojevi (tj.: 3.14 kao 314) te se koristi Newton-Raphsonova metodu za dobivanje n-tog korijena od d iteracija rješavanjem jednadžbe $xn - d = 0$.

Na taj način se također može prilagoditi cijena operacije prema potrebnoj preciznosti. Rezultati su pokazali da je cijena izračuna kvadratnog korijena 23209gwei te 31304gwei za izračun potpune opracije podudaranja (oko 0,000031 ETH ili 0,00527

USD s tadašnjom cijenom ETH).

Iako se to na prvi pogled može činiti malom vrijednošću, ali u stvarnom biometrijskom sustavu operacija podudaranja morala bi se ponoviti stotine ili tisuće puta. Na primjer, za scenarij s 10.000 korisnika, ukupni trošak za operaciju podudaranja (autentifikacija svakog korisnika u sustavu) bio bi nešto veći od 50 USD s tadašnjim cijenama.

Isječak pametnog ugovora za Euklidsku udaljenost, te n-ti korijen:

```
function templateMatching(uint threshold, uint _dp,
        uint[] memory baseArray, uint[] memory userArray) public
        ↪  view returns(bool) {
        uint distance = 0;

        for (uint i = 0; i < array1.length; i++) {
                distance += (baseArray[i] - userArray[i])**2;
        }

        return nthRoot(distance, 2, _dp, 1000) <= threshold;
}

function nthRoot(uint _a, uint _n, uint _dp, uint _maxIts) pure
↪  public returns(uint) {
        assert (_n > 1);

        uint one = 10 ** (1 + _dp);
        uint a0 = one ** _n * _a;

        uint xNew = one;
        uint x = 0;
        uint t0 = 0;

        uint iter = 0;
```

```
        while (xNew != x && iter < _maxIts) {
                x = xNew;
                t0 = x ** (_n - 1);
                if (x * t0 > a0) {
                        xNew = x - (x - a0 / t0) / _n;
                } else {
                        xNew = x + (a0 / t0 - x) / _n;
                }
                ++iter;
        }

        return (xNew + 5) / 10;
}
```

4.1.2 Hammingova udaljenost

Nasuprot Euklidskoj udaljenosti, Hammingovu udaljenost između korisničkih ključeva *hash* tablice znatno je lakše implementirati te je gotovo besplatna u ekonomskom smislu.

Svaka operacija implementirana na *blockchain*-u mora biti optimizirana što je više moguće. Korišten je algoritam čije vrijeme izvođenja ovisi o broju jedinica prisutnih u binarnom obliku zadanog broja postižući mnogo bolju izvedbu [33].

Za razliku od euklidske udaljenosti, ovaj izračun udaljenosti je jednostavniji i koristi samo bitovne operacije. Osim toga, ne pohranjuje podatke na *blockchain* te ima dodatanu prednost da operacija čitanja ne zahtijeva dodatne troškove. Ovo je posebno važno u biometrijskom scenariju, gdje veliki broj korisnika može biti uključen u operacije podudaranja.

Isječak pametnog ugovora za Hammingovu udaljenost:

```
function HammingDistance(int a, int b) pure public returns(int) {

        // Get A XOR B...
```

```
        int c = a ^ b;

        // and now count the number of ones
        int count=0;
        while (c!=0)
        {
                // This works because if a number is power of 2,
                // then it has only one 1 in its binary
                 ↪  representation.
                c = c & (c-1);
                count++;
        }

        return count;
}
```

4.1.3 Ekonomska cijena

Pohranjujući milijun predložaka za detekciju lica koštao bi 740000 dolara za takav sustav s punim načinom pohrane na *blockchain*.

Shema sa pohranom sažetaka značajno bi poboljšala te brojke, jer ne pohranjuje same podatke, već samo njihov sažetak koji jamči integritet. Za isti scenarij, trošak bi bio mnogo razumniji iznos od 14800 dolara.

Shema Merkleovih stabala podrazumijevala bi trošak od samo jednog centa dolara (0,0148 USD) za pohranu bilo koje količine predložaka. Nadalje, i operacija izmjene predloška imala bi isti trošak. Međutim, čak i za biometrijski sustav koji radi u velikoj korporaciji ili okruženju, ti se troškovi čine razumnim.

Sve te cijene mogu uvelike varirati ovisno o cijeni ETH, koja, kao i ostale kriptovalute, obično trpi oštra povećanja i padove cijene.

4.2 Osiguranje biometrijskog autentifikacijskog sustava korištenjem blockchaina

Prema [34] BDAS je novi biometrijski sustav autentifikacije koji koristi blockchain. Uključivanjem blockchain tehnologije, BDAS pruža decentralizirani i distribuirani mehanizam za biometrijsku autentifikaciju bez oslanjanja na središnji autentifikacijski modul. U BDAS-u svaki klijent neovisno upravlja fragmentima biometrijskog predloška i obrađuje operaciju provjere autentičnosti.

Ovaj decentralizirani mehanizam omogućuje robusnu autentifikaciju eliminirajući rizike kvarova koji nastaju oslanjanjem na centralni entitet.

Štoviše, svaki je predložak podijeljen na fragmente i tim fragmentima upravljaju različiti klijenti. Ovaj mehanizam segmentacije omogućuje sigurno upravljanje biometrijskim podacima minimiziranjem rizika od curenja cijelog predloška. S obzirom da zbog nepromjenjivosti biometrijskih podataka, curenje biometrijskih podataka može uzrokovati dugoročne sigurnosne prijetnje.

U osnovi, temeljne operacije BDAS-a, uključujući upravljanje predlošcima i aktivnosti snimanja autentifikacije, implementirane su u pametni ugovor koji se izvodi na blockchainu.

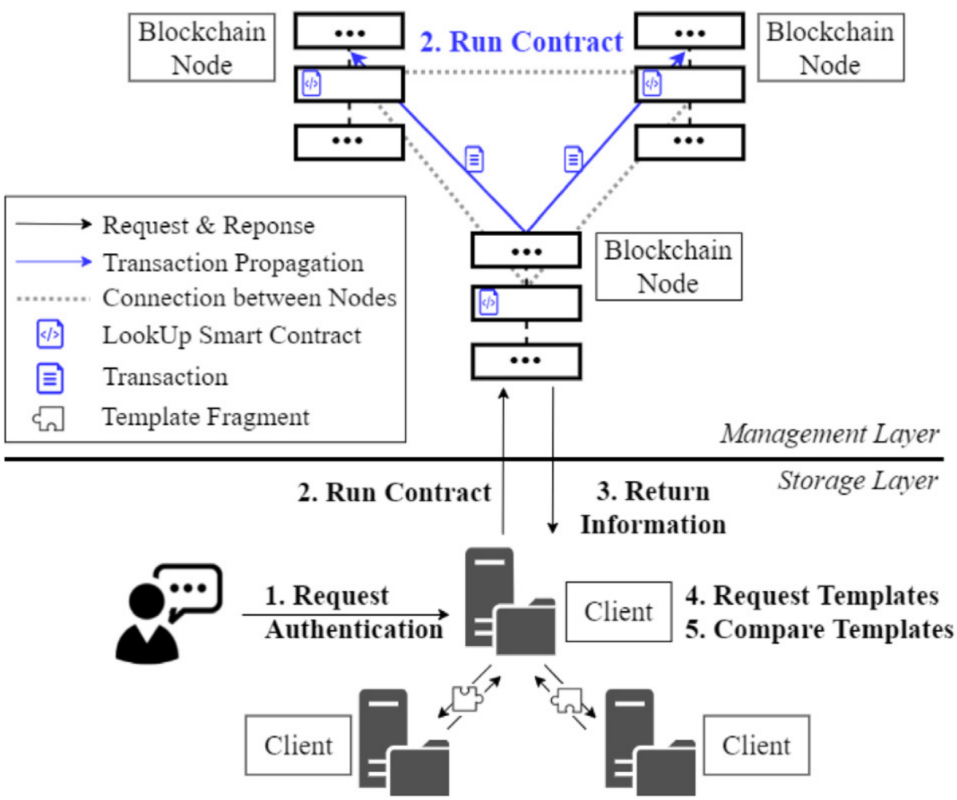
Kako bi se osigurala sigurnost koju pruža distribuirano upravljanje informacijama, najmanje tri klijenta moraju konfigurirati BDAS, a svaki klijent u BDAS-u pokreće jedan blockchain čvor. BDAS je implementiran kao dopušteni blockchain u kojem mogu sudjelovati samo verificirani čvorovi.

Sastoji se od dva sloja: sloja za pohranu i sloja za upravljanje. U sloju za pohranu, klijenti obrađuju zahtjeve korisnika, a u sloju upravljanja, čvorovi *blockchain*-a konfiguriraju mrežu.

Svaki klijent samostalno obrađuje autentifikaciju i upravlja fragmentima predložaka, što se razlikuje od rada postojećih sustava autentifikacije baziranih na poslužitelju i klijentu.

Pokretanjem pametnog ugovora klijent pronalazi odgovarajuće klijente koji upravljaju potrebnim fragmentima.

Svaki proces autentifikacije u BDAS-u se bilježi kao transakcija na *blockchain* mreži.



Slika 4.1 Pregled BDAS operacija [34]

4.3 Polygon ID

Predstavlja rješenje na Polygon mreži od samih kreatora Polygona koje postiže skalabilni model za digitalne identitete bez dopuštenja i otporne na cenzuru. Iako nije direktno rješenje za biometrijsku autentifikaciju, s obzirom na prirodu autentifikacije projekta i korištenja *zero knowledge* tehnologije, može poslužiti kao baza za razvoj biometrijskog sustava.

Projekt je u cijelosti otvorenog koda.

Polygon ID je rješenje za identifikaciju koje korisnicima omogućuje korištenje *zero knowledge* tehnologije za interakciju s pametnim ugovorima, na temelju provjerljivih akreditivnih isprava izdanih izvan lanca [35].

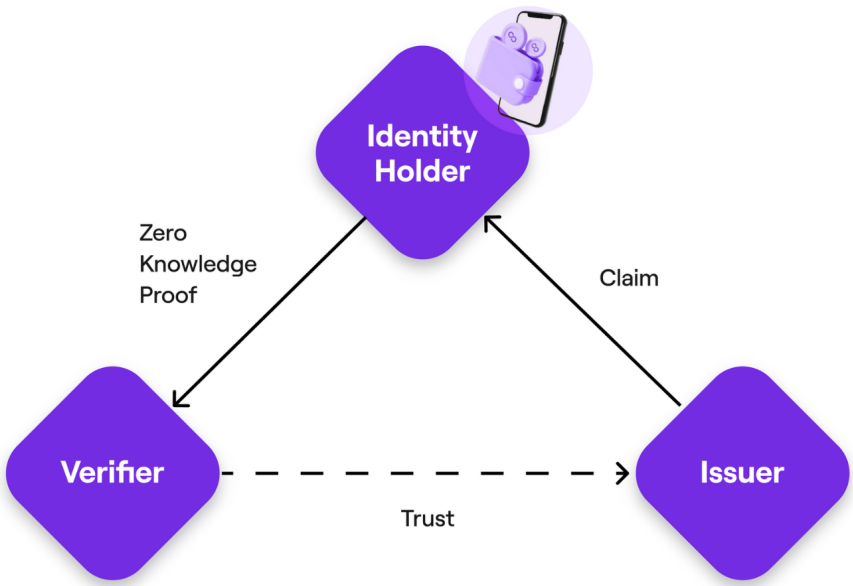
Polygon ID zadovoljava *Verifiable Credentials* i *Decentralized Identifiers* W3C standarde, omogućujući povećanje interoperabilnosti decentraliziranih aplikacija.

Pojedinci primaju i pohranjuju zahtjeve kao što je KYC ček u osobnom novčaniku i koriste *zero knowledge* tehnologiju za privatnu provjeru izjava o njima. Može sigurno komunicirati s pametnim ugovorima i drugim identitetima bez otkrivanja osobnih podataka.

Prema 4.2 arhitektura se sastoji od tri modula:

1. *Identity Holder*
2. *Issuer*
3. *Verifier*

Ovo troje zajedno čini ono što nazivaju trokutom povjerenja. Verifikacija se može odraditi *on-chain* ili *off-chain*.



Slika 4.2 Polygon ID trokut povjerenja [36]

4.4 Cardano Midnight

Cardano Midnight je infrastruktura usredotočena na privatnost koja funkcionira kao *sidechain* Cardano *blockchain*-a.

Midnight je namijenjen zaštiti osjetljivih poslovnih i osobnih podataka. Koristi *zk-SNARKs* tehnologiju, tehnologiju koja štiti privatne podatke i transakcije koje su izvršili korisnici.

Zahvaljujući protokolu Kachina [37], zaštita podataka koju pruža Midnight omogućit će stvaranje decentraliziranih aplikacija sposobnih za zaštitu podataka svojih korisnika unutar blockchaina.

Projekt je još u razvoju te kod nije javno dostupan, stoga nije moguće bez uvida u kod utvrditi ispravnost tvrdnji projekta te potencijalne prednosti u integraciji s biometrijom.

Poglavlje 5

Predloženo konceptualno rješenje

5.1 Komponente sustava

5.1.1 zkEVM

Zero-Knowledge Ethereum Virutal Machine (zkEVM) je virtualni stroj koji generira zero-knowledge dokaze za provjeru ispravnosti programa. zkEVM-ovi su dizajnirani za izvršavanje pametnih ugovora na način koji podržava zero-knowledge tehnologiju.

Oni su dio zkEVM rollupova, rješenja za skaliranje Ethereum sloja 2 koji poboljšavaju propusnost prijenosom računanja i pohrane stanja izvan lanca. Šalju podatke o transakcijama Ethereumu zajedno s zero-knowledge dokazima dokazujući valjanost skupina transakcija izvan lanca.

Kao i EVM, zkEVM je virtualni stroj koji prelazi između stanja kao rezultat programskih operacija. Ali zkEVM ide i dalje izrađujući dokaz koji potvrđuje točnost svakog dijela izračuna. U biti, zkEVM koristi mehanizam za dokazivanje da su koraci izvršavanja slijedili predodređena pravila [38].

Način rada zkEVM

zkEVM generira zero-knowledge dokaze provjeru različitih elemenata u svakom dijelu rada:

1. Pristup bajt-kodu: Je li odgovarajući programski kod ispravno učitan, s prave adrese?
2. Operacije čitanja i pisanja: a. Je li program dohvatio prave vrijednosti iz stoga/memorije/skladišta prije izračuna? b. Je li program zapisao ispravne izlazne vrijednosti u stog/memoriju/pohranu nakon završetka izvođenja?
3. Izračun: Jesu li operacijski kodovi ispravno izvedeni (tj. jedan za drugim, bez preskakanja koraka)?

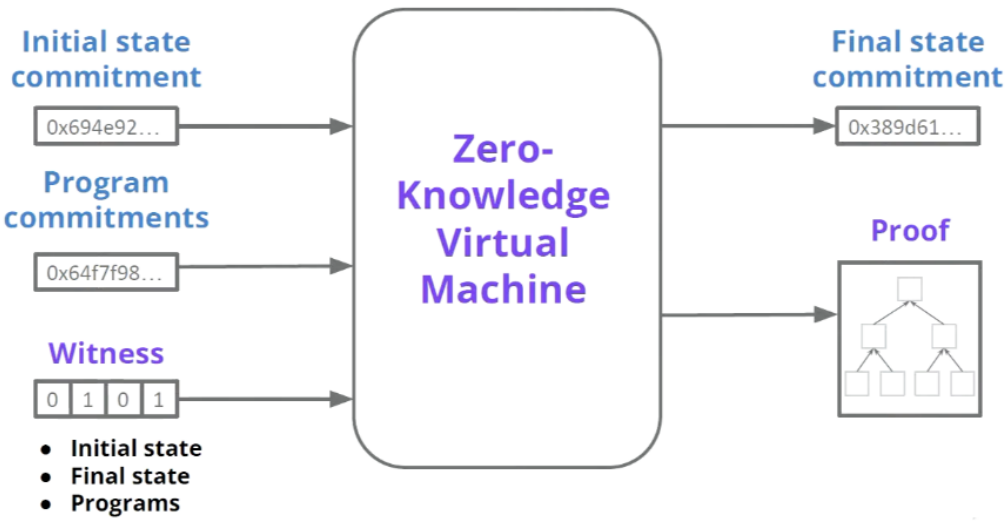
zkEVM je podijeljen u tri dijela: okruženje za izvođenje, krug za provjeru i ugovor s verifikatorom. Svaka komponenta doprinosi izvršavanju programa zkEVM-a, stvaranju dokaza i provjeri dokaza.

1. izvršno okruženje je mjesto gdje se pokreću programi (pametni ugovori) u zkEVM-u. Izvršno okruženje zkEVM-a funkcionira poput EVM-a: potrebno je početno stanje i trenutna transakcija za izlaz novog (konačnog) stanja.
2. Krug za dokazivanje proizvodi zero-knowledge dokaze koji provjeravaju valjanost transakcija izračunatih u izvršavajućem okruženju. Proces generiranja dokaza dovršava se upotrebom predstanja, ulaza transakcije i informacija nakon stanja kao ulaza. Nakon toga, dokazivač dobiva jezgrovit dokaz o valjanosti tog određenog prijelaza stanja.
3. zero-knowledge rollupovi podnose dokaze valjanosti pametnom ugovoru na L1 lancu (Ethereum) na provjeru. Ulaz (predstanja i informacije o transakciji) i izlaz (konačna stanja) također se dostavljaju ugovoru ovjeritelja. Zatim verifikator pokreće izračun na pruženom dokazu i potvrđuje da su dostavljeni izlazi ispravno izračunati iz ulaza.

5.1.2 Multi party computation

Protokoli za multiparty computation (MPC) omogućuju skupu stranaka $(p_1, p_2, ...p_n)$ da međusobno komuniciraju te izračunaju zajedničku funkciju $(F(d_1, d_2, ...d_n))$ svojih privatnih ulaza $(d_1, d_2, ...d_n)$ ne otkrivajući pritom ništa osim izlaza funkcije.

Potencijalne primjene za MPC su ogromne: dražbe koje čuvaju privatnost, pri-



Slika 5.1 Generiranje dokaza zkEVM-a [39]

vatne usporedbe DNK, privatno strojno učenje, i više.

Zbog ovoga, MPC je intenzivna tema istraživanja u akademskoj zajednici otkad je predstavljen u 1980-ima za dvostranački slučaj (FOCS 1986) [40], te Goldreich, Micali i Wigderson za višestranački slučaj (STOC 1987) [41].

Nedavno je MPC postao dovoljno učinkovit da se može koristiti u praksi, te je napravio prijelaz od objekta teorijskog proučavanja do tehnologije koja se koristi u industriji [42].

Model i korištenje MPC-a u praksi

Stvarna paradigma za definiranje sigurnosti zapravo ima neke vrlo važne implikacije za korištenje MPC-a u praksi. Konkretno, da bi se koristio MPC protokol, sve što treba učiniti je razmotriti sigurnost svog sustava kada osoba od povjerenja provodi proračun za koji se koristi MPC. Ako je sustav u ovom slučaju siguran, onda će ostati siguran čak i kada se koriste pravi MPC protokoli. Znači da se ne mora razumjeti bilo što o tome kako MPC protokoli rade ili čak kako je definirana sigurnost. Idealan

model pruža čistu i lako razumljivu apstrakciju.

Bilo koji ulazi su dozvoljeni

Iako paradigma stvarnog modela pruža jednostavnu apstrakciju, kao što je gore opisano, postoji suptilna točka koja je ponekad pogrešno shvaćena. U praksi, suparničke strane mogu unijeti bilo koje vrijednosti koje žele, i doista ne postoji generički način da se to spriječi. Dakle, ako dvoje ljudi želi vidjeti tko ima veću plaću (bez otkrivanja više od ove jedne informacije), onda ništa ne sprječava jednog od njih da unese maksimalnu moguću vrijednost kao svoju plaću (a zatim se pošteno ponašati u MPC-u samog protokola), s rezultatom da zarađuju više no što zapravo zarađuju.

Dakle, ako sigurnost aplikacije ovisi o tome koristi li strana točne unose, tada se moraju koristiti mehanizmi koji to uzimaju u obzir. Na primjer, moguće je zahtijevati potpisane unose i provjeriti potpis kao dio MPC izračuna. Ovisno o specifičnom protokolu, to može značajno povećati troškove.

MPC osigurava proces, ali ne i izlaz

Još jedna suptilnost koja se često pogrešno shvaća je da MPC osigurava proces, što znači da se ništa ne otkriva samim proračunom. Međutim, to ne znači da izlaz funkcije koja se izračunava ne otkriva osjetljive informacije. Za ekstremni primjer, zamislimo dvoje ljudi koji računaju prosjek svojih plaća. Doista je istina da ništa osim prosjeka neće biti izlaz, međutim pojedinac, s obzirom na svoju plaću i prosjeka obiju plaća, može izračunati točnu plaću druge osobe. Tako, samo korištenje MPC-a ne znači da su svi problemi s privatnošću riješeni. Umjesto toga, MPC osigurava proces računanja, te pitanje zbog čega bi se funkcije trebale, a koje ne bi trebale računati i dalje se treba pozabaviti pitanjima privatnosti.

Shamirovo dijeljenje tajne

MPC protokoli obično koriste dijeljenje tajni kao osnovni alat. Stoga je bitno shvaćati Shamirove sheme dijeljenja tajni [43]. Shema dijeljenja tajni rješava problem trgovca koji želi podijeliti tajnu između sebe i n strana, tako da bilo koji podskup od $t + 1$

ili više strana može rekonstruirati tajnu, ali nijedan podskup od t ili manje strana može saznati bilo što o tajni. Shema koja ispunjava ove zahtjeve naziva se shema dijeljenja tajni $(t+1)$-od-n.

Shamirova shema dijeljenja tajni koristi činjenicu da za bilo koju $t+1$ točku na dvodimenzionalnoj ravnini $(x_1, y_1), \ldots, (x_{t+1}, y_{t+1})$ s jedinstvenim x_i, postoji jedinstveni polinom $q(x)$ stupnja najviše t tako da je $q(x_i) = y_i$ za svaki i. Nadalje, moguće je učinkovito rekonstruirati polinom $q(x)$, ili bilo koje specifične točke na njemu. Jedan od načina da se to učini je pomoću Lagrangeove baze polinoma $\ell_1(x), \ldots, \ell_t(x)$, gdje se rekonstrukcija provodi računanjem $q(x) = \sum_{i=1}^{t+1} \ell_i(x) \cdot y_i$. Odavde ćemo pretpostaviti da su svi proračuni u konačnom polju $\mathbb{Z}_p$, za prost $p > n$.

S obzirom na gore navedeno, kako bi podijelio tajnu s, djelitelj odabire slučajni polinom $q(x)$ stupnja najviše t pod ograničenjem da $q(0) = s$. Konkretno, djelitelj postavlja $a_0 = s$ i bira slučajne koeficijente $a_1, \ldots, a_t \in \mathbb{Z}_p$, i postavlja $q(x) = \sum_{i=0}^{t} a_i \cdot x^i$.). Tada, za svaki $i = 1, \ldots, n$, trgovac daje i-toj strani udio $y_i = q(i)$; to je razlog zašto nam je potrebno da vrijedi $p > n$, dakle da se svakoj stranci mogu dati različiti udjeli. Rekonstrukcija prema podskupu bilo kojih t stranki je moguća jednostavnom interpolacijom polinoma za izračunavanje $q(x)$ i zatim izvođenjem uz $s = q(0)$. Iako $t+1$ stranke mogu potpuno oporaviti s, nije teško pokazati da bilo koji podskup od t ili manje strana ne može naučiti nešto o s. To je zbog činjenice da imaju t ili manje točaka na polinomu, i tako postoji polinom koji prolazi kroz ove točke i točku $(0, s)$ za sve moguće $s \in \mathbb{Z}_p$. Nadalje, budući da je polinom slučajan, svi polinomi su jednako vjerojatni, pa tako su i sve vrijednosti $s \in \mathbb{Z}_p$ jednako vjerojatne.

MPC poštene većine s tajnim dijeljenjem

Prvi korak u većini protokola za opći MPC (tj. protokoli koji se mogu koristiti za izračunavanje bilo koje funkcije) jest predstavljanje funkcije koja se izračunava kao Booleov ili aritmetički sklop. U slučaju MPC poštene većine na temelju dijeljenja tajni, aritmetički sklop (koji se sastoji od množenja i adicijskih vrata) je nad konačnim poljem $\mathbb{Z}_p$ s $p > n$, kao i gore. Napomena, kako su aritmetički sklopovi Turing potpuni, svaka se funkcija može prikazati u tom obliku. Strane koje sudjeluju u MPC protokolu sve imaju ovaj krug i pretpostavljamo da svi mogu komunicirati sigurno

jedan s drugim.

5.1.3 Kvantna sigurnost

Jedan od problema asimetrične enkripcije, koja se obično koristi za našu svakodnevnu komunikaciju pomoću sigurnih web-preglednika, chatova, VPN-ova i tako dalje, jest ranjivost od kvantnih računala. Oslanja se na privatni i javni ključ koji su matematički povezani, pri čemu je javni ključ odgovoran za kodiranje ili provjeru. Privatni ključ namijenjen je samo određenom entitetu koji dekriptira ili potpisuje podatke.

Mnogi asimetrični kriptografski algoritmi oslanjaju se na matematički problem koji se zove rastavljanja brojeva na proste faktore, a što je ključ duži – što više bitova sadrži – to je teže dekriptirati upotrebom iscrpnog pretraživanja. I dok današnja računala ne mogu razbiti te algoritme, kvantno računalo bi moglo – zahvaljujući Shorovom algoritmu [44], koji je razvio Peter Shor 1994. To je zato što je rastavljanje brojeva na proste faktore, bez obzira na to koliko je dug niz, dječja igra za kvantno računalo s milijunima od kubita.

Danas se kripto protokoli kao što su SSL, Transport Layer Security i HTTPS temelje na takozvanim kriptografskim "primitivima" - kriptografskim algoritmima niske razine. To uključuje digitalne potpise, sheme provjere autentičnosti i sheme šifriranja, ali ti protokoli postaju beskorisni ako su kripto primitivi ugroženi.

Tu može pomoći rešetkasta kriptografija [45]. Oslanja se na područje matematike nazvano "geometrija brojeva", gdje su podaci skriveni unutar rešetki, složenih algebarskih struktura. Iako je lako stvoriti točku u prostoru koja je blizu rešetke, zahtjevnost kriptografije temeljene na rešetki je ta što je teško ići u suprotnom smjeru. Pronalaženje najbližeg mjesta u rešetki iz točke u prostoru zahtijeva vrijeme koje je eksponencijalno u dimenziji rešetke.

Ovaj se problem proučava od 1970-ih i učinkovit algoritam za njega imao bi mnoge primjene u mnogim važnim područjima. Također je dobio veliku pozornost zajednice kvantnih algoritama.

5.1.4 Homomorfna enkripcija

Homomorfna enkripcija je pretvorba podataka u šifrirani tekst koji se može analizirati i raditi s njim kao da je još uvijek u izvornom obliku. Takav oblik enkripcije omogućuje izvođenje složenih matematičkih operacija na šifriranim podacima bez ugrožavanja enkripcije [46].

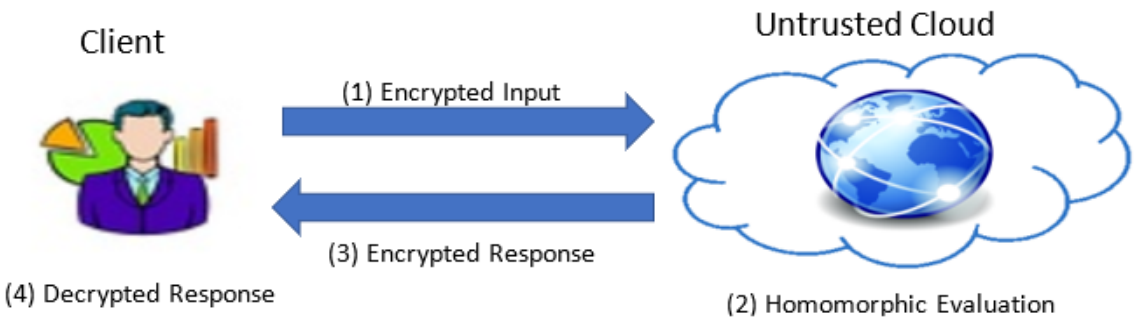
U matematici, homomorfizam opisuje transformaciju jednog skupa podataka u drugi uz očuvanje odnosa između elemenata u oba skupa.

Izraz je izveden iz grčkih riječi za istu strukturu. Budući da podaci u homomorfnoj shemi šifriranja zadržavaju istu strukturu, identične matematičke operacije dat će ekvivalentne rezultate - neovisno o tome izvodi li se akcija na šifriranim ili dešifriranim podacima.

Razlikuje se od uobičajenih metoda enkripcije jer omogućuje izvođenje matematičkih izračuna izravno na šifriranim podacima, što trećim stranama predstavlja sigurnije kao i privatnije rukovanje korisničkim podacima.

Potpuna homomorfna enkripcija se razlikuje od shema djelomične homomorfne enkripcije, gdje se samo jedna vrsta operacija može izvesti na šifriranim podacima (šifrirani tekst), na primjer, množenje ili zbrajanje. Potpuno homomorfne sheme šifriranja mogu podržati operacije množenja, i množenje i zbrajanje, omogućujući procjenu više operacija preko šifriranog teksta. Kod potpune homomorfne enkripcije omogućen je beskonačan broj zbrajanja ili množenja šifriranih tekstova. Programi za bilo koju funkcionalnost mogu se pokrenuti na šifriranim ulazima kako bi proizveli šifrirani izlaz.

Slika 5.2 prikazuje tipičnu FHE primjenu. Klijentova ulazna poruka, obično u obliku vektora, prvo se kodira u polinom otvorenog teksta, koji se zatim šifrira u šifrirani tekst koji se sastoji od dva polinoma, na primjer, $c = (a, b)$, od N cjelobrojnih koeficijenata modulo neki veliki cijeli broj Q.
Nakon izvođenja izračuna šifriranog teksta homomorfno, bilo putem lokalnog ili putem udaljenog oblaka, korisnik prima šifrirani odgovor, a zatim dekriptira i dekodira odgovor kako bi saznao rezultat izračuna.
Podaci i izračun ostaju šifrirani tijekom cijelog procesa [47].



Slika 5.2 Jednostavan primjer povjeravanja računanja nepouzdanoj trećoj strani koristeći potpuno homomorfnu enkripciju [47]

Ukratko, potpuna homomorfna enkripcija pruža programerima moćan alat za rješavanje pitanja privatnosti i sigurnosti u prethodno opisanim situacijama.

Unatoč svim prednostima, danas nije u širokoj upotrebi. Ostaje veliki jaz u izvedbi između rada na šifriranim podacima i njima u izvornom obliku. Za izračun za koji bi bila potrebna jedna sekunda trebalo bi više od 11 dana da se izvrši korištenjem trenutnih homomorfnih knjižnica za šifriranje kao što su HElib ili PALISADE. Ovo usporavanje od oko milijun puta neprihvatljiv je kompromis te je time homomorfna enkripcija neizvediva u praksi i znatno ograničena u primjeni.

5.2 Opis konceptualnog rješenja

Predloženo konceptualno rješenje inspirirano je postojećim rješenjima iz 4 te nadograđeno metodama i tehnologijama opisanim u radu.

Takav sustav bi koristio IPFS za pohranu biometrijskih podataka - bilo značajki i/ili predložaka, zavisno o implementaciji, dok bi se na blockchain putem nezamjenjivih tokena povezale sve željene informacije u obliku metapodataka. Time bi integritet informacija bio sačuvan na samome blockchainu.

Bilo tko bi tada mogao pribosti metapodatke nezamjenjivog tokena s interpla-

netarnog datotečnog sustava, odnosno specificirati da se podaci zadrže i postoje na jednom ili više IPFS čvorova.

Programabilnost blockchaina, odnosno pametni ugovori mogu se iskoristiti prilikom izrade validnih tokena - analogno upisu podataka u bazu, za usporedbu predložaka i izvješćivanje sudionika o samome procesu. Takvi ugovori bi se po mogućnosti izvršavali na mrežama sloja 2 za dodatne benefite privatnosti kao što je skrivanje procesa izvršavanja ugovora na mreži. Zbog manjih troškova i sami nezamjenjivi tokeni nalazili bi se na zk mreži sloja 2.

Ukoliko je moguće, koristila bi se homomorfna enkripcija za biometrijske podatke. Dodatan potencijalni benefit bilo bi i korištenje kvantno sigurnih algoritama za enkripciju.

Korisnici bi preporučljivo same uzorke prikupljali izvan samog sustava za koji se autentificiraju, naravno ukoliko je to moguće. Takvo što moglo bi se koristiti ukoliko je korisniku dostupna pripadajuća infrastruktura, npr. kamera ili senzor otiska prsta na mobitelu i slično. Ukoliko je potrebno, izrada nezamjenjivog tokena mogla bi se vršiti pod nadzorom, kako bi se osigurao identitet učesnika sustava.

Moguće je ukomponirati i MPC ako bi se jedan predložak pohranjivao kod više učesnika sustava ili pak ukoliko bi se provodila identifikacije umjesto autentifikacije, odnosno provjere jednog predloška sa svima sa svrhom određivanja ukoliko je korisnik prisutan u sustavu.

5.2.1 Shema sustava

Shema predloženog sustava 5.3 sastoji se od nekoliko komponenti. Smjer strelica prikazuje tok podataka i komunikacijske kanale. Interplanetarni datotečni sustav na kojem su pohranjeni biometrijski podaci. Blockchaina i pripadajućeg rješenja sloja 2. Zbog jednostavnosti, sam blockchain i njegova pripadajuća zk mreža sloja 2 prikazani su kao jedna spojena cjelina. Učesnici su u vlasništvu nezamjenjivih tokena, koje kreiraju putem pametnih ugovora. Biometrijske značajke i/ili uzorke učesnici sustava mogu kreirati samostalno ili uz pomoć centralnog sustava. Centralni sustav predstavlja srž ovakvog rješenja te se za njega odrađuje autentifikacija, kreiranje,

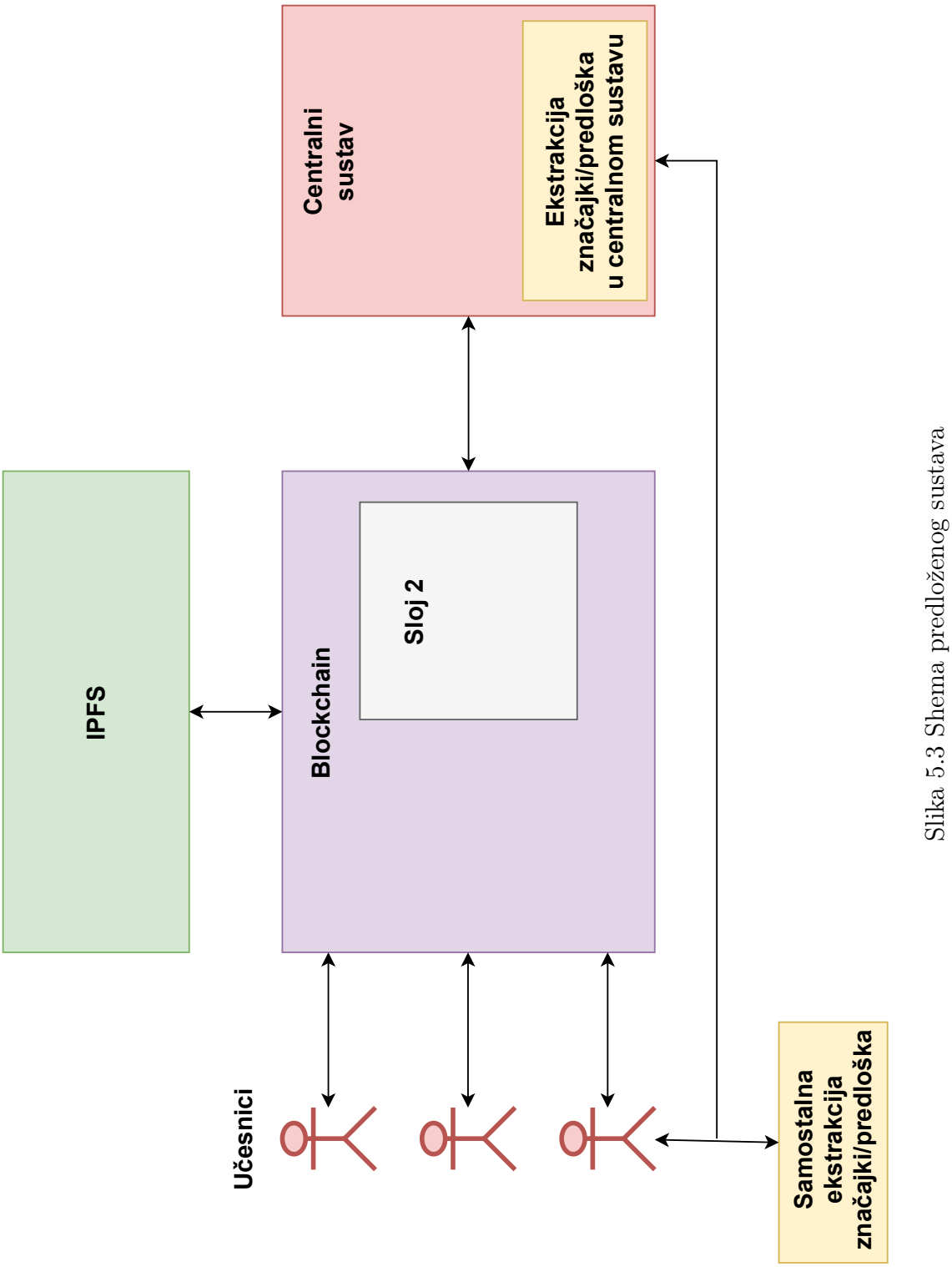
brisanje i izmjena nezamjenjivih token - uz pristanak učesnika ovisno o operaciji. Listu trenutnih članova sustava za autentifikaciju sam sustav trebao bi pohraniti na blockchain.

5.2.2 Upis u sustav

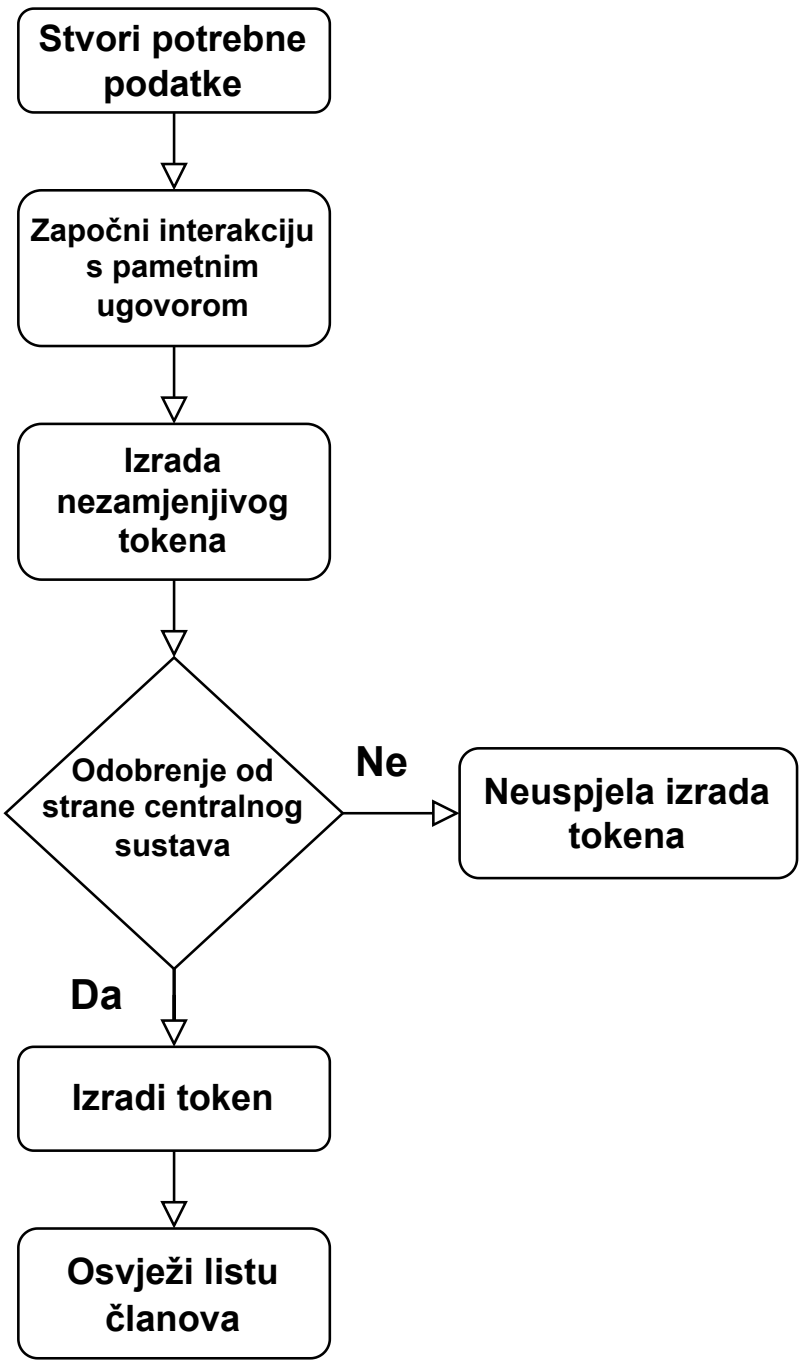
Upis u sustav započinje svaranjem potrebnih biometrijskih podataka, ovisno o odabranom izgledu nezamjenjivog tokena. Pilikom te faze vrši se ekstrakcija biometrijskog uzorka i stvaranje biometrijskog predloška. Ekstrakcija uzorka može biti samostalna ili u sklopu centralnog sustava. Nakon uspješnog prikupljanja podataka započinje interakcija s pametnim ugovorom za izradu samog nezamjenjivog tokena. Ukoliko centralni sustav ne odobri korisniku izradu tokena, ista je neuspjela. Pri uspješnoj izradi tokena, učesniku se stvara nezamjenjivi token na adresi s kojom je imao interakciju s pametnim ugovorom. Lista članova biometrijskog sustava se osvježava - dodaje se novi učesnik.

5.2.3 Autentifikacija

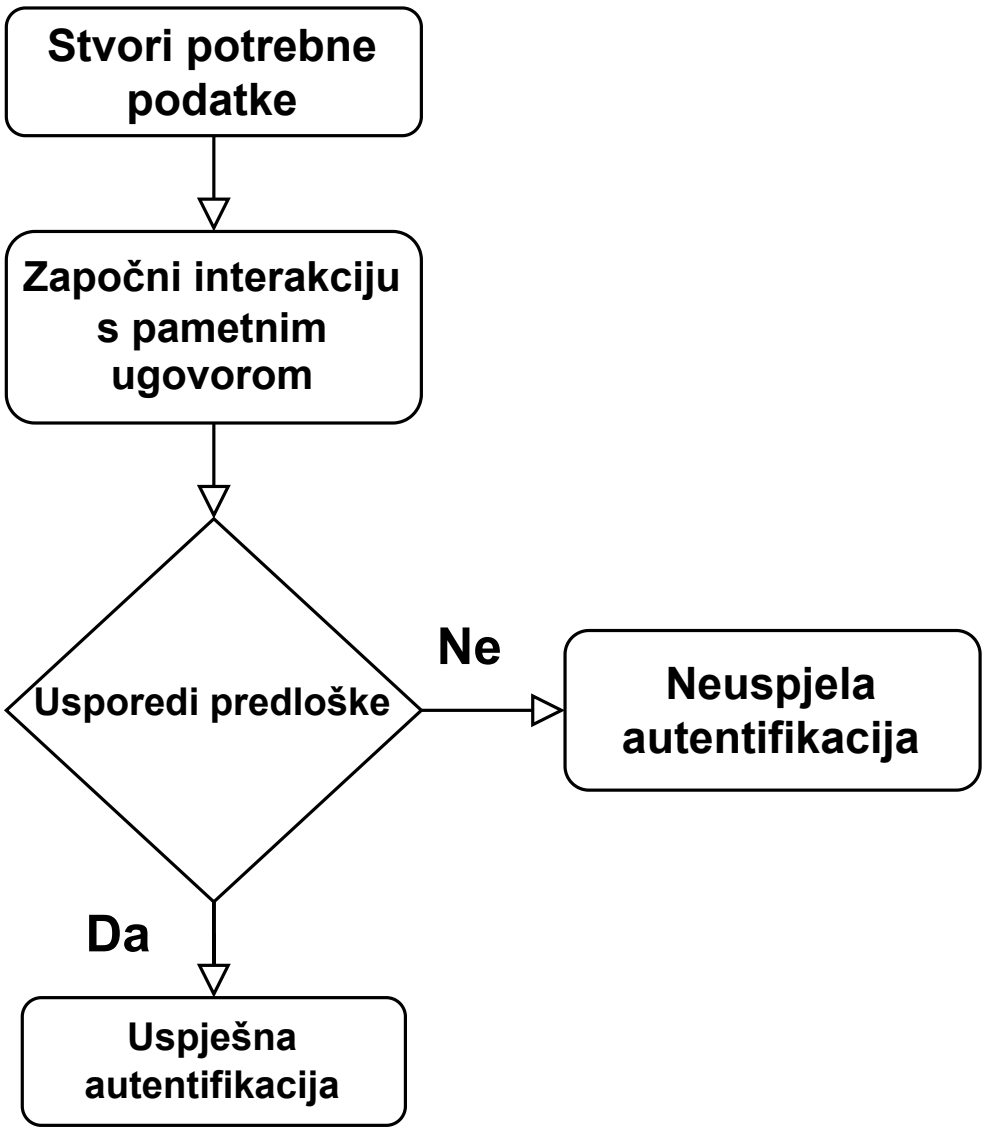
Autentifikacija započinje stvaranjem potrebnih biometrijskih podataka, te se stvara biometrijski predložak korisnika. Ovaj proces bilo bi potrebno odraditi pod nadzorom kako ne bi došlo do lažnog predstavljanja. Korisnik započinje interakciju s pametnim ugovorom. Ukoliko korisnik nije onaj za kojeg se predstavlja, odnosno nije učesnik u sustavu, autentifikacija je neuspješna. U suprotnom autentifikacija je uspješna te je korisnika zapravo i učesnik u sustavu.



Slika 5.3 Shema predloženog sustava



Slika 5.4 Dijagram toka upisa



Slika 5.5 Dijagram toka autentifikacije

5.3 Razmatrane Blockchain Mreže

Razmatrane blockchain mreže jesu mreže generacije 2 i 3; odnosno mreže koje podržavaju korištenje pametnih ugovora s obzirom na eliminaciju posrednika, odnosno dobivanje pouzdanog programabilnog posrednika u obliku virtualnog stroja na *blockchainu*.

S obzirom na popularnost, implementirane funkcionalnosti te prethodno iskustvo, odabrani su Ethereum i Cardano.

Iako implementiraju nezamjenjive tokene na različite načine, te koriste drukčije modele računa, itd. oboje pružaju tehnologije i ekosustav za implementaciju predloženog modela.

5.3.1 Ethereum

Ethereum je tehnologija za izgradnju aplikacija i organizacija, držanje digitalne imovine, transakcije i komunikaciju bez kontrole središnjeg tijela.

Nema potrebe predavati sve svoje osobne podatke da bi se koristio Ethereum - korisnici imaju kontrolu nad svojim podacima i onim što se dijeli. Ethereum ima vlastitu kriptovalutu Ether koja se koristi za plaćanje određenih aktivnosti na Ethereum mreži.

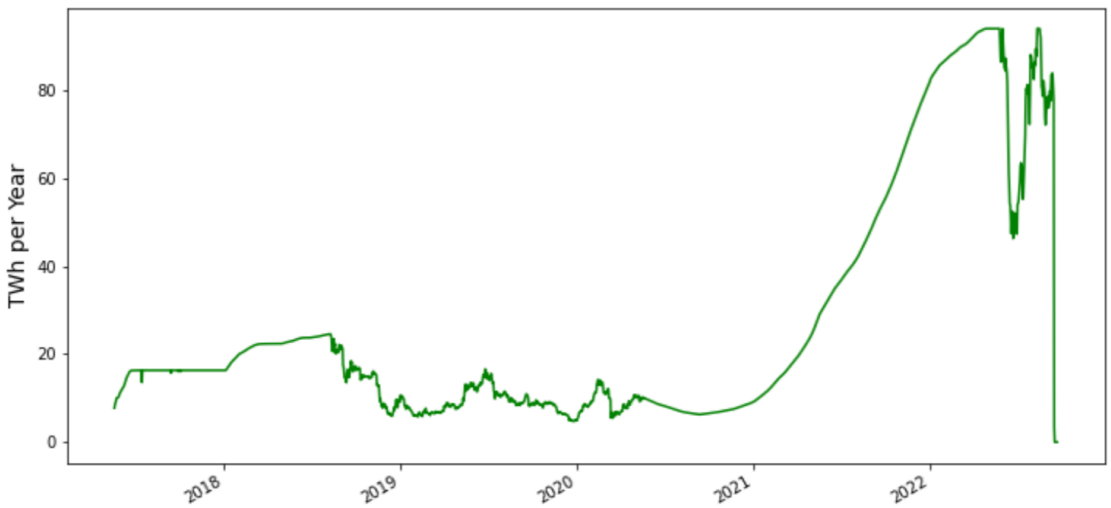
Ethereum je programabilan, što omogućuje izgradnju i implementaciju decentraliziranih aplikacije na njegovoj mreži.

Programabilnost Ethereuma predstavlja mogućnost izrade aplikacija koje koriste blockchain za pohranu podataka ili kontrolu onoga što vaša aplikacija može učiniti. To rezultira blockchainom opće namjene koji se može programirati da radi bilo što. Budući da nema ograničenja za ono što Ethereum može učiniti, to omogućuje širok spektar opcija na Ethereum mreži.

Dok je Bitcoin samo mreža za plaćanje, Ethereum je više poput tržišta financijskih usluga, igara, društvenih mreža i drugih aplikacija koje poštuju privatnost i slobodu korištenja [48].

Ethereum blockchain započeo je s radom 30. srpnja 2015. Više od sedam godina,

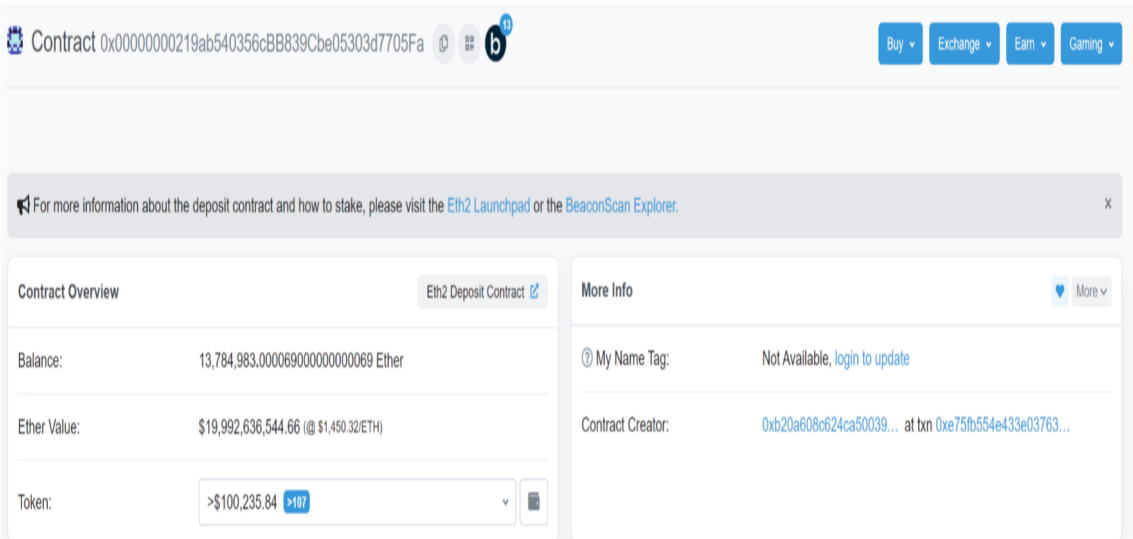
blockchain je koristio konsenzus mehanizam proof-of-work (POW). 15. rujna 2022. Ethereum mreža usvojila je konsenzus mehanizam proof-of-stake (POS). Prijelaz na POS je smanjio potrošnju energije za 99,98% [49].



Slika 5.6 Potrošnja energije u Ethereum mreži [49]

Na grafu 5.6 moguće je vidjeti nagli pad u potrošnji mreže pred kraj 2022. što predstavlja prelazak na POS konsenzus mehanizam.

Blokove sada osiguravaju validatori, sudionici u konsenzus mehanizmu Ethereum 2.0 koji su stavili najmanje 32 ETH u depozitni ugovor prikazan na slici 5.7.



Slika 5.7 Ethereum 2.0 depozitni ugovor [50]

Odbor od najmanje 128 validatora, koje je odabrao RANDAO, biraju su za dodavanje bloka za bilo koji *slot*. Jedan sudionik koji se naziva predlagač bloka formira blok, proces koji uključuje odabir i provjeru da skup transakcija nema kvarova ili pogrešaka. Blok dakle trebaju potvrditi preostali validatori, zvani atesteri, koji provjeravaju i daju svoje glasove o povjerenju u blok. Konačnost se postiže na bloku kontrolne točke, prvom bloku u sljedećoj epohi, uz podršku od 2/3 uloženog ETH [51].

Slučajnost odabira znači da validatori moraju imati gotovo aktivan sustav 24/7; može im se spaliti dio ili cijeli ulog i biti uklonjen (presječen) sa skupa validatora, zbog neispunjavanja dodijeljenog zadatka.

Zaključno s 2022.11.07., 218 validatora je imalo dio svog uloga spaljen [52].

5.3.2 Cardano

Cardano je POS programabilna blockchain platforma: prva koja se temelji na recenziranim istraživanjima i razvija pomoću metoda utemeljenih na znanstvenim dokazima. Kombinira pionirske tehnologije za pružanje neusporedive sigurnosti i održivosti decentraliziranim aplikacijama, sustavima i društvima.

Ouroboros je prvi dokazano siguran POS protokol i prvi blockchain protokol koji se temelji na recenziranom istraživanju.

Ouroboros kombinira jedinstvenu tehnologiju i matematički provjerene mehanizme kako bi osigurali sigurnost i održivost blockchaina koji o njemu ovise.

Rezultat je protokol s dokazanim sigurnosnim jamstvima koji može olakšati širenje globalnih mreža bez dopuštenja s minimalnim energetskim zahtjevima [53].

U središtu Ouroborosa je koncept beskonačnosti. Globalne mreže moraju biti u stanju rasti održivo i etički: kako bi svijetu pružile veće mogućnosti, a istovremeno ga očuvale [54].

Zajamčeno je da je protokol siguran sve dok 51% udjela – u slučaju Cardana, Ada – drže pošteni sudionici, što se, uz druge nove koncepte, postiže nasumičnim odabirom voditelja. Protokol se nastavlja razvijati kroz nove iteracije i rigoroznu sigurnosnu analizu.

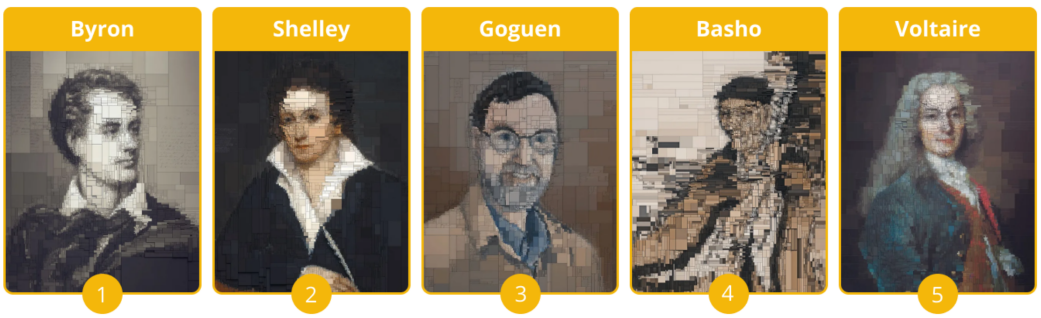
Cardanu je nedostajao autoritativni *white paper* koji bi unaprijed odredio njegovu poziciju unutar kripto svijeta. Umjesto toga, osnivač, Charles Hoskinson je zamislio izgradnju slojeva sustava za rješavanje problema koji postoje unutar drugih kripto ekosustava.

S tim ciljem na umu, Hoskinson je osnovao tri entiteta - Cardano Foundation, Input Output (IOHK) i EMURGO - za razvoj blockchain ekosustava.

Cardano *roadmap* sažetak je razvoja Cardana koji je organiziran u pet epoha: Byron, Shelley, Goguen, Basho i Voltaire. Svaka je era usredotočena na skup funkcionalnosti koje će se isporučivati kroz više izdanja koda.

Ere su imena dobile po značajnim intelektualcima, a mogu se vidjeti na 5.8.

Dok će se ere Cardano blockchaina isporučivati uzastopno, rad za svaku eru odvija



Slika 5.8 Pet era Cardano blockchaina [55]

se paralelno, s istraživanjem, izradom prototipova i razvojem koji su često u tijeku istovremeno.

Radovi svakog razdoblja prikupljeni su i predstavljeni javno.

5.4 Osnovni pristupi pohrane podataka

U osnovi postoje tri glavna pristupa za pohranu podataka na blockchainu. Kako bi se spriječila njegova zlouporaba, trošak prostora za pohranu posebno je skup u usporedbi s računanjem. Stoga je ključno je procijeniti takav trošak, kako bi se sveo na minimum, odabirom optimalne sheme pohrane [33].

5.4.1 Full on-chain storage

Svi podaci pohranjeni su takvi kakvi jesu na blockchainu. Na primjer, biometrijski predlošci se mogu izravno pohraniti kao podatkovne strukture putem pametnih ugovora.

To je najjednostavnija shema i stoga, najskuplja i najneučinkovita.

Kao primjer, 5.1 prikazuje trošak čitanja i pisanja 1KB podataka na Ethereum, gdje se *gas* odnosi na jedinicu koja mjeri količinu računalnih resursa potrebnih za izvršavanje određenih operacija, odnosno predstavlja naknadu potrebnu za izvršenje

transakcija na samoj mreži.

Tablica 5.1 Nepromjenjivi troškovi pohrane na Ethereumu. 1 gwei = 10^{-9} ETH, i 1 ETH = 1209 USD (u vrijeme pisanja, studeni 2022.)

Operation	Gas/KB	ETH/KB	$/KB
READ	6,400	0.000032	$0.03869
WRITE	640,000	0.0032	$3.8688

5.4.2 Data hashing

Učinkovitiji pristup je da se podatci pohranjuju izvan samog blockchaina dok se on koristi samo kao jamstvo cjelovitosti zbog svoje intrinzične nepromjenljivosti sažetka.

Ovako, umjesto cijelog podatka, samo je njegov *hash*, odnosno sažetak pohranjen na blockchain (kroz pametne ugovore).

Zatim se kompletan predložak može pohraniti u bilo koji drugi tradicionalni vanjski sustav za pohranu.

Kako bi se održao duh distribuiranosti, otpornost na cenzuru i visoka dostupnost blockchaina, u ovom bi slučaju bilo poželjno koristiti sustave distribuirane pohrane kao što su IPFS ili Swarm [56].

Jedan nedostatak ovog pristupa je taj što je i dalje je potrebno osigurati dostupnost podataka pohranjenih izvan blockchaina. Ako su takvi podatci izgubljeni ili neovlašteno mijenjani, čak i kada bi se ta izmjena uvijek primijetila putem različitih vrijednosti sažetaka, održivost sustava bila bi ugrožena.

5.4.3 Vezane strukture podataka

Shemu s hashiranjem može se dodatno poboljšati upotrebom povezane strukture podataka, npr. Merkleova stabla [57].

Ova specifična struktura podataka naširoko se koristi u domeni kriptografije i računarstva kao što su provjera integriteta baze podataka, peer-to-peer mreže i blockchain [58].

Merkleovo stablo je binarna struktura podataka u kojoj svaki čvor sadrži kriptografski hash njegovih ulančanih podređenih čvorova.

Zbog ovog rekurzivnog načina konstruiranja, korijen stabla sadrži informacije o ostalim čvorovima, a modifikacija bilo kojeg sadržaja čvora će uzrokovati potpunu promjena vrijednosti korijena.

Poglavlje 6

Implementacija

Biometrijski podaci jedinstveni su i kritični podatci pojedinaca, kao takvi ne bi trebali biti sadržani u jednoj velikoj bazi čije bi kompromitiranje moglo dovesti do gubitka svih biometrijskih podataka.

S obzirom na dugovječnost pojedinca te važnost biometrijskih podataka u životu istog, svaki pojedinac bi trebao biti sam u vlasništvu svojih podataka, jedno rješenje takvoj problematici mogu biti nezamjenjivi tokeni.

U nastavku, biti će prikazano mintanje NFTa na Cardano blockchainu sa biometrijskim metapodacima koji bi se kasnije mogao koristiti unutar biometrijskog sustava.

6.1 Okruženje

Programski kod testiran je na Linux distribuciji Arch Linux, kernelu `linux-hardened 6.0.7`. Korišteni *shell* je Bourne Again Shell (BASH). Sklopovlje se sastoji od `Thinkpad P14s` s `AMD Ryzen 5850U`, te `32GB RAM`.

6.2 IPFS čvor

Instaliranje IPFS-a putem komandne linije zgodno je ukoliko se planira graditi aplikacije i usluge na temelju IPFS čvora.

Korištenje IPFS-a putem komandne linije omogućuje iskorištavanje svega što IPFS Desktop aplikacija može, ali na detaljnijoj razini jer je moguće granularno odrediti koje naredbe treba izvršiti [59] .

Instalacija alata:

```
# pacman -S kubo
```

Testiranje ispravne instalacije:

```
$ ipfs --version
> ipfs version 0.16.0
```

`ipfs` pohranjuje sve svoje postavke i interne podatke u direktorij koji se zove repozitorij. Prije prve uporabe IPFS-a, potrebno je inicijalizirati ga sljedećom naredbom [60]:

```
$ ipfs init
```

Za pridruživanje svog čvora javnoj mreži, potrebno je pokrenuti IPFS servis na drugom terminalu i pričekati da se pokažu sva tri donja retka kako bi čvor bio spreman za uporabu [61]:

```
$ ipfs daemon

> Initializing daemon...
> API server listening on /ip4/127.0.0.1/tcp/5001
> Gateway server listening on /ip4/127.0.0.1/tcp/8080
```

6.3 Instalacija Nix upravitelja paketa

Nix je čisto funkcionalni upravitelj paketa koji ima za cilj učiniti upravljanje paketima pouzdanim i ponovljivim [62] [63].

Za instalaciju upravitelja paketa:

```
# pacman -S nix
```

Pokretanje pozadinskog servisa.

```
# systemctl enable --now nix-daemon.service
```

Dodavanje željenog `$USER` u `nix-users` grupu.

```
# groupadd nix-users
# usermod -aG your_user
```

Dodavanje Nix kanala:

```
$ nix-channel --add https://nixos.org/channels/nixpkgs-unstable
$ nix-channel --update
```

6.4 Cardano čvor na testnoj mreži

Cardano čvor biti će izgrađen iz izvornog koda pomoću `nix` upravitelja paketa [64] [65]. Kako bi se ubrzao proces izgradnje, moguće je koristiti cache izvršnih datoteka održavan od strane Input Output Hong Kong (IOHK). Ovaj korak je opcionalan, no vrijeme izgradnje binarnih datoteka se znatno ubrzava.

```
# sudo mkdir -p /etc/nix
# cat <<EOF | sudo tee -a /etc/nix/nix.conf
substituters = https://cache.nixos.org https://cache.iog.io
trusted-public-keys =
 ↪  hydra.iohk.io:f/Ea+s+dFdN+3Y/G+FDgSq+a5NEWhJGzdjvKNGv0/EQ=
 ↪  cache.nixos.org-1:6NCHdD59X431o0gWypbMrAURkbJ16ZPMQFGspcDShjY=
EOF
```

Ponuđene su dvije testne mreže, naime *Preview* i *Preprod*.

S obzirom da se ovaj rad bavi istraživanjem a ne pripremom za glavnu mrežu, koristiti će se *Preview* testna mreža.

Za instalaciju punog čvora:

```
$ git clone https://github.com/input-output-hk/cardano-node
$ cd cardano-node
$ git checkout 1.35.5
$ nix build .\#cardano-node -o cardano-node-build
```

Sinkronizacija s testnet mrežom potrajala je oko 2 sata s obzirom da se sa stare testne mreže nedavno prešlo na dvije navedene mreže. Preporuča se koristiti najnoviju verziju čvora i alata komandne linije s obzirom na navedene promjene. U trenutku pisanja, to je `v1.35.5`.

Za instalaciju `cardano-cli`:

```
$ nix build .#cardano-cli -o cardano-cli-build
$ ./cardano-cli-build/bin/cardano-cli
```

Za lakše korištenje, izrađujemo direktorij s konfiguracijama i direktorij za bazu:

```
$ mkdir -p test/db
```

Nakon pozicioniranja u direktorij, potrebno je preuzeti konfiguracijske datoteke [66]:

```
curl -O -J         https://book.world.dev.cardano.org/environments/p⌋
↪  review/config.json
curl -O -J         https://book.world.dev.cardano.org/environments/p⌋
↪  review/db-sync-config.json
curl -O -J         https://book.world.dev.cardano.org/environments/p⌋
↪  review/submit-api-config.json
curl -O -J         https://book.world.dev.cardano.org/environments/p⌋
↪  review/topology.json
```

```
curl -O -J         https://book.world.dev.cardano.org/environments/p⌋
↪  review/byron-genesis.json
curl -O -J         https://book.world.dev.cardano.org/environments/p⌋
↪  review/shelley-genesis.json
curl -O -J         https://book.world.dev.cardano.org/environments/p⌋
↪  review/alonzo-genesis.json
```

Za jednostavniji razvoj, postavljaju se određene *shell* varijable okruženja, moguće ih je ubaciti u `~/.bashrc` ukoliko je korišteni `shell bash` ili u zasebnu skriptu te ju potom `source`-ati.

```
#!/usr/bin/env bash # za kompatibilnost u *nixevima

export PATH=$PATH:/putanja/do/cardano-src/cardano-node/cardano-node-⌋
↪  build/bin/
export PATH=$PATH:/putanja/docardano-src/cardano-node/cardano-cli-bu⌋
↪  ild/bin/

export CARDANO_NODE_SOCKET_PATH=/home/dndrej/src/cardano-src/cardano⌋
↪  -node/state-node-testnet/node.socket
```

Stanje sinkronizacije s testnetom može se pratiti s [67]:

```
$ watch cardano-cli query tip --testnet-magic 2
```

Nakon završetka sinkronizacije, izlaz će biti varijacija sljedećeg:

```
watch cardano-cli query tip $CARDANO_NODE_TESTNET

{
        "block": 540788,
        "epoch": 139,
        "era": "Babbage",
        "hash": "19fb812b75f3dfd04625c17f0a88d935b039c898db5df9cce18⌋
        ↪  b5d573e762c30",
        "slot": 12089276,
        "syncProgress": "100.00"
}
```

6.5 Prijenos podataka na IPFS

Prikvačivanje je vrlo važan koncept u IPFS-u. IPFS semantika pokušava stvoriti osjećaj da je svaki pojedini objekt lokalni — ne postoji "preuzmi ovu datoteku za mene s udaljenog poslužitelja", samo `ipfs cat` ili `ipfs get`, koji djeluju na isti način bez obzira gdje se stvarni objekt nalazi.

Ponekad je potrebno imati mogućnost kontrole nad objektima, odnosno lokalnu pohranu. Prikvačivanje je mehanizam koji omogućuje da se kaže IPFS-u da uvijek zadrži dani objekt negdje — zadan je lokalni čvor, iako to može biti drugačije ako se koristi uslugu daljinskog prikvačivanja na neki od servisa treće strane [68]. IPFS ima prilično agresivan mehanizam predmemoriranja koji će zadržati objekt lokalnim neko vrijeme nakon što se izvrši bilo koja IPFS operacija nad njime, ali ti se objekti mogu redovito skupljati u smeće. Kako bi se spriječilo to skupljanje smeća, jednostavno se može prikvačiti content identifier (CID) objekta [69].

Datoteke ili direktoriji se dodaju na IPFS putem:

```
$ ipfs add <path>
```

6.6 Stvaranje biometrijskog nezamjenjivog tokena

Kao što naziv kaže - mora biti "nezamjenjiv". To znači da treba imati jedinstvene identifikatore ili atribute kako bi se razlikovalo od drugih.

Većinom NFT-evi bi trebali živjeti na lancu zauvijek. Stoga nam treba neki mehanizam koji će osigurati da NFT ostane jedinstven i da se ne može duplicirati.

6.6.1 PolicyID

Nezamjenjivi tokeni na Cardanu imaju sljedeće karakteristike:

1. Iznos/vrijednost (koliko ih ima?)
2. Ime
3. jedinstveni policyID

Budući da nazivi nezamjenjivih tokena nisu jedinstveni i mogu se lako duplicirati, Cardano NFT-evi moraju biti identificirani policyID.

Ovaj ID je jedinstven i trajno povezan sa tokenom. On proizlazi iz skripte pravila koja definira karakteristike kao što su tko i kako može stvarati nove tokene te kada se te radnje mogu napraviti.

Mnogi NFT projekti čine policyID pod kojim su NFT-evi stvoreni javno dostupnim, tako da svatko može razlikovati lažne/duplicirane NFT-eve od originalnih tokena.

Neke usluge nude registraciju policyID-a za otkrivanje tokena koji imaju iste atribute kao i određeni token, ali su izrađeni prema drugim pravilima.

6.6.2 Metapodaci

Osim jedinstvenog policyID-a, transakciji je moguće priložiti i metapodatke s različitim atributima.

Primjer s s NFT Maker-a [70].

```
{
  "721": {
    "{policy_id}": {
      "{policy_name}": {
        "name": "<required>",
        "description": "<optional>",
        "sha256": "<required>",
        "type": "<required>",
        "image": "<required>",
        "location": {
          "ipfs": "<required>",
          "https": "<optional>",
          "arweave": "<optional>"
        }
      }
    }
  }
}
```

Metapodaci nam pomažu da prikažemo stvari poput URI-ja slika i stvari koje ga uistinu čine NFT-em. Na ovaj načina platforme kao što je pool.pm [71] mogu lako pratiti NFT-eve do posljednje transakcije, čitati metapodatke i tražiti slike i atribute u skladu s istim.

Trenutno ne postoji dogovoreni standard o tome kako se NFT ili metapodaci definiraju. Međutim, postoji prijedlog za poboljšanje Cardana [72].

Također se uključuje putanja do `cardano-cli` unutar `PATH` varijable okruženja kako bi `shell` znao gdje pronaći izvršnu datoteku prilikom poziva naredbe.

Moguće je provjeriti ukoliko su sve variajble okruženja dobro postavljene sljedećom naredbom:

```
$ env | grep -E '^PATH=|^CARDANO_NODE_SOCKET_PATH='
```

6.6.3 Radni direktorij

Potrebno je kreirati novi direktorij te se pozicionirati u isti:

```
$ mkdir nft
$ cd nft/
```

6.6.4 Postavljanje varijabli

Postavljaju se određene varijable za bolju čitljivost i otklanjanje pogrešaka neuspjelih transakcija.

Od verzije `cardano-node 1.31.0` naziv tokena trebao bi biti u hex formatu. Postavit ćemo varijablu $realtokenname (pravo ime u utf-8) i zatim je pretvoriti u $tokenname (ime u hex formatu).

```
$ realtokenname="NFT1"
$ tokenname=$(echo -n $realtokenname | xxd -b -ps -c 80 | tr -d '\n')
$ tokenamount="1"
$ fee="0"
$ output="0"
$ ipfs_hash="please insert your ipfs hash here"
```

6.6.5 Izrada ključeva i adrese

Za izradu ključeva:

```
$ cardano-cli address key-gen --verification-key-file payment.vkey
↪  --signing-key-file payment.skey
```

Isti se mogu iskoristiti za izradu plaćajuće adrese:

```
$ cardano-cli address build --payment-verification-key-file
↪  payment.vkey --out-file payment.addr --testnet-magic 2
```

Sažetak adrese spremamo u varijablu:

```
$ address=$(cat payment.addr)
```

6.6.6 Testni tokeni

S obzirom da transakcije uvijek zahtijevaju određenu količinu tokena, potrebno ih je pribaviti, putem npr. slavine testne mreže [73].

Stanje adrese može se provjeriti pomoću:

```
$ cardano-cli query utxo --address $address --testnet-magic 2
```

6.6.7 Parametri protokola

Za izračune transakcija potrebni su neki od trenutnih parametara protokola. Parametri se mogu spremiti u datoteku pod nazivom `protocol.json` ovom naredbom:

```
$ cardano-cli query protocol-parameters --testnet-magic 2 --out-file
↪  protocol.json
```

6.6.8 Izrada policyID-a

Za generiranje `policyID` potrebno je kreirati ključeve te skriptu pravila.

```
$ mkdir policy
```

Izrada ključeva:

```
cardano-cli address key-gen                     \
    --verification-key-file policy/policy.vkey \
    --signing-key-file policy/policy.skey
```

Definiramo skriptu koja omogućuje izradu pomoću samo jednog ključa, te izradu ili spaljivanje tokena unutar `10000` slotova od transakcije.

izrađuje se `policy.script` prema sljedećem predlošku:

```
{
  "type": "all",
  "scripts":
  [
    {
      "type": "before",
      "slot": <insert slot here>
    },
    {
      "type": "sig",
      "keyHash": "insert keyHash here"
    }
  ]
}
```

6.6.9 Izrada transakcije

Prije izrade transakcije potrebno je postaviti okruženje. Za parametre adrese:

```
cardano-cli query utxo --address $address $CARDANO_NODE_TESTNET
```

Parametre iz izlaza naredbe potrebno je pohraniti u varijable:

```
txhash="insert your txhash here"
txix="insert your TxIx here"
```

```
funds="insert Amount in lovelace here"
policyid=$(cat policy/policyID)
output=1400000
```

Izlazna vrijednost postavlja se na 1400000 Lovelace što je ekvivalentno 1.4 ADA. Ovaj se iznos koristi jer je to minimalni iznos zahtjevan od strane UTxO-a.

Nakon postavljanja svih varijabli:

```
cardano-cli transaction build \
--testnet-magic 2 \
--alonzo-era \
--tx-in $txhash#$txix \
--tx-out $address+$output+"$tokenamount $policyid.$tokenname" \
--change-address $address \
--mint="$tokenamount $policyid.$tokenname" \
--minting-script-file $script \
--metadata-json-file metadata.json  \
--invalid-hereafter $slotnumber \
--witness-override 2 \
--out-file matx.raw
```

Ukoliko naredba generira izlaz kao što je:

```
Minimum required UTxO: Lovelace 1448244
```

Potrebno je promijeniti vrijednost varijable `$output` na dobivenu vrijednost.

Ako je minimalna vrijednost bila točna, ova naredba će generirati `matx.raw` i dati izlaz sličan ovome:

```
Estimated transaction fee: Lovelace 176677
```

Potrebno je potpisati transakciju:

```
cardano-cli transaction sign  \
```

```
--signing-key-file payment.skey  \
--signing-key-file policy/policy.skey  \
--testnet-magic 2 --tx-body-file matx.raw  \
--out-file matx.signed
```

Preostaje podnijeti transakciju, dakle izraditi nezamjenjivi token:

```
cardano-cli transaction submit --tx-file matx.signed --testnet-magic
↪  2
```

Nezamjenjivi token bi sada trebao biti izrađen. Sadržaj adrese može se provjeriti putem:

```
cardano-cli query utxo --address $address --testnet-magic 2
```

6.6.10 Biometrijski token na testnoj mreži

Nezamjenjivi token je stvoren na `Preview` testnoj mreži.

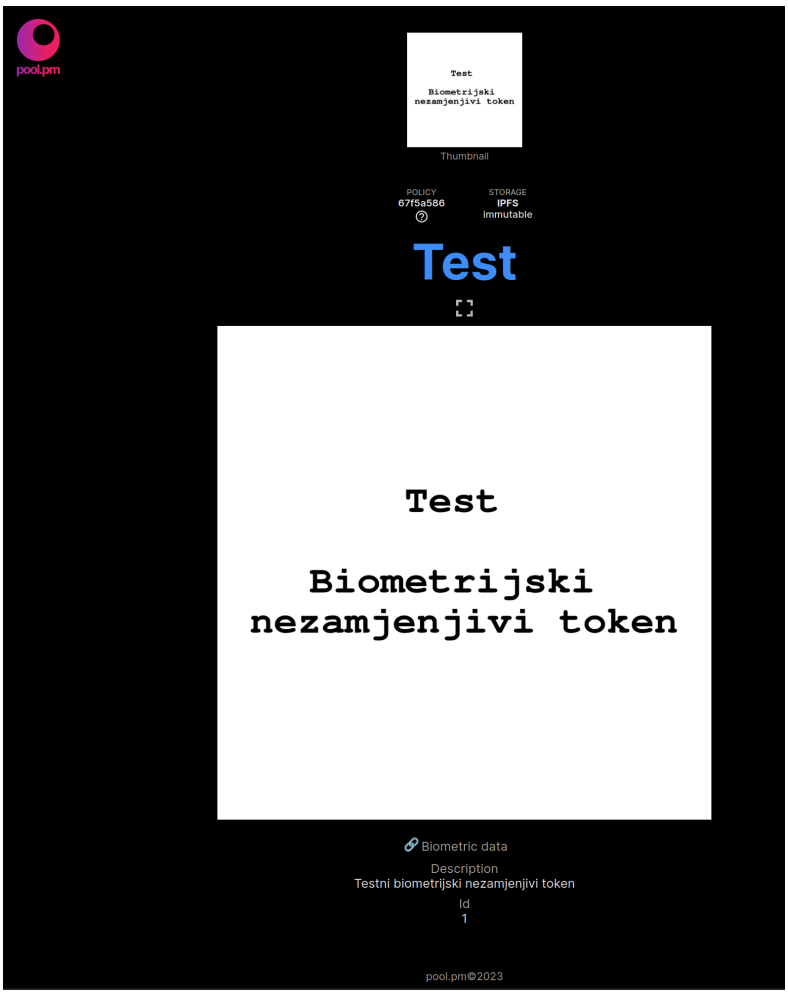
Polje `image` predstavlja IPFS sažetak na sliku nezamjenjivog tokena, dok polje `biometric_data` predstavlja generično polje u svrhu ilustracije pohrane biometrijskih podataka na IPFS, te naposljetku i u nezamjenjivi token.

Metapodaci tokena su sljedeći:

```
{
    "721": {
      "67f5a586753fb7fc0b5e8ec485a140d772eb146d370559f219e0400b": {
              "test_nft": {
                      "description": "Testni biometrijski
                       ↪  nezamjenjivi token",
                          "name": "Test",
                          "id": "1",
                          "image": "ipfs://Qma4B5TuuFMfrmfjQdo8CmrEH⌋
                           ↪  ouX9nKpScEnXhzsDKRZkq",
```

```
                                "biometric_data": "ipfs://QmVkFifkHXRb9Xsj⌋
                                ↪  AKrHm1K4xU5XZEZZTidQSEzAdq3nFP"
                        }
                }
        }
}
```

Izgled nezamjenjivog tokena prema `pool.pm` nakon stvaranja je sljedeći:



Slika 6.1 pool.pm prikaz metapodataka stvorenog primjera tokena

U `policy.script` definirano je stvaranje tokena u rasponu od trenutka stvaranja do `1000 slot` unaprijed za pripadajući `policyID`. Sadržaj datoteke `policy.script` je sljedeći:

```
{
    "type": "all",
    "scripts":
    [
    {
        "type": "before",
        "slot": 12091406
    },
    {
        "type": "sig",
        "keyHash": "3975227f0cb438b9a6efe26e8a5863aeda4aeed764ff38f2
        ↪  6ce37c2c"
    }
    ]
}
```

Adresa na kojoj je stvoren nezamjenjivi token je `addr_test1vr8ykt8svtpknejax6aylcgnhkxlvvjtcqkm5r4rm3qhnvgz26e58`. Detalje o adresi, uključujući i nezamjenjivi token moguće je provjeriti na `Cardanoscan`-u za `preview` testnu mrežu [74].

Poglavlje 7

Diskusija

Predloženi koncept sustava za decentraliziranu biometrijsku autentifikaciju ima prednost raspršivanja nezamjenjivih tokena po učesnicima sustava, iako to može biti i mana uvođenjem uspješnog malicioznog napadača koji dobiva pristup korisničkim biometrijskim predlošcima. Takav problem nastoji se minimizirati nadzorom prilikom autentifikacije korisnika.

Kada su korisnici u vlasništvu vlastitih podataka imaju moć odlučivanja gdje ih koristiti te se time mogu odlučiti za privatnija rješenja.

Dodatna decentralizacija ostvaruje se uvođenjem stvaranja tokena putem vlastitog čvora uz interakciju s pametnim ugovorom, bez oslanjanja na već postojeće API-e. Takav način stvaranja tokena daje više mogućnosti, ekonomično je isplatljiviji, te pridodaje veću pouzdanost i privatnost sustavu. Uz korištenje vlastitog čvora, postoje i opcije uporabe TOR mreže i sličnih rješenja za privatnost i anonimnost [75].

Pomoću nezamjenjivih tokena u vlasništvu svakog od učesnika se također minimizira gubitak podataka kod uvođenja malicioznih napdača s pristupom centraliziranoj bazi. Takva shema također sa sobom nosi i dodatnu odgovornost na korisnicima, s obzirom da prilikom gubitka ili ukradenog privatnog ključa gubi biometrijske podatke. Novnoastali problem mogao bi se riješiti dijeljenjem biometrijskih predložaka u fragmente kao što je učinjeno u riješenju 4.2.

S obzirom na prirodu implementacije nezamjenjivih tokena, njihova sigurnost, a time indirektno i privatnost korisnika mogu biti ugroženi. U radu se može zami-

jetiti kako su nezamjenjivi tokeni na Cardano mreži nativni, dok se na Ethereum mreži implementiraju kroz pametne ugovore. Cardano mreža može biti i zanimljiva iz perspektive formalne specifikacije i verifikacije s obzirom na izbor jezika za implementaciju protokola i pametnih ugovora.

Također treba obratiti pozornost na centralizaciju rješenja sloja 2 i `side-chain`-ova. Bilo bi preporučljivo imati redundantna riješenja u slučaju pada mreže ili sličnih problema.

S obzirom na javno izvršavanje pametnih ugovora, bilo bi dobro naginjati prema Zero-Knowledge (zk) rješenjima koja skrivaju proces a pružaju samo dokaz te time ne cure kontekst ili dodatne informacije.

Iskorištavanjem Polygon sloja 2 na Ethereum *blockchain* mreži za izvršavanje pametnih ugovora moguće je smanjiti troškove za 90% [76]. Što bi moglo znatno pogurati neke već postojeće ekonomično neisplative ali validne sustave.

Poglavlje 8

Zaključak

Cilj ovog rada bio je proučiti i predložiti model privatne pohrane biometrijskih podataka na *blockchainu*.

Predloženo je konceptualno rješenje za privatnu autentifikaciju koja koristi nezamjenjive tokene, odnosno nezamjenjive tokene za pohranu sažetka biometrijskih podataka na *blockchain* u obliku metapodataka, osiguravajući njihov integritet, dok je sama pohrana enkriptiranih podataka realizirana pomoću IPFS-a.
Odabran je navedeni pristup s obzirom na decentraliziranost samih podataka, odnosno njihovo raspršivanje po korisnicima, kako bi se dodatno zaštitila privatnost istih. Privatnost prilikom autentifikacije i same izrade nezamjenjivog tokena je poboljšana korištenjem zk tehnologije i potencijalno, homomorfne enkripcije.

Predloženo je korištenje post-kvantno sigurnih enkripcijskih metoda kako bi se zaštitili korisnički podaci s obzirom na relativnu dugovječnost pojedinca u današnjici.

Odabrane *blockchain* mreže su Ethereum te Cardano, s obzirom na podršku pametnih ugovora, popularnost, opsežnu dokumentaciju, prethodno iskustvo, itd. Također razlike u implementaciji nezamjenjivih tokena i tehnologija na kojima su izrađene mreže služe kao zanimljive teme za diskusiju.

Izvršavanje takvog sustava moglo bi se odvijati na samom 1. sloju *blockchaina*, no preporuča se korištenje 2. sloja blockchaina, odnosno sloja koji koristi Zero-Knowledge (zk) tehnologiju za dodatan dobitak u području privatnosti, brzini izvršavanja te ekonomskoj isplativosti.

Bibliografija

[1] i. S. P. A. K. Jain, A. Ross, *An introduction to biometric recognition.* IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no. 1, pp. 4–20,, 2004.

[2] E. B. Budish. The Economic Limits of Bitcoin and Anonymous, Decentralized Trust on the Blockchain. , s Interneta, https://ssrn.com/abstract=4148014 2022.

[3] S. P. M. E. K. i. C. Y. Deepak Puthal, Nisha Malik. The Blockchain as a Decentralized Security Framework [Future Directions]. , s Interneta, https://doi.org/10.1109/mce.2017.2776459 2018.

[4] P. O. E. Glen Weyl and V. Buterin. Decentralized Society: Finding Web3's Soul. , s Interneta, https://ssrn.com/abstract=4105763 2022.

[5] F. M. D. C. T. Y. N. V. i. C. J. T. Carlo Campajola, Raffaele Cristodaro. The Evolution Of Centralisation on Cryptocurrency Platforms. , s Interneta, https://arxiv.org/abs/2206.05081 2022.

[6] Y. W. i. X. Z. Lin William Cong, Ke Tang. Inclusion and Democratization Through Web3 and DeFi? Initial Evidence from the Ethereum Ecosystem. , s Interneta, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4162966 2022.

[7] T. E. D. P. Supervisor. Opinion 4/2018 on the proposals for two regulations establishing a framework for interoperability between eu large-scale information systems. , s Interneta, https://www.peercoin.net/whitepapers/peercoin-paper.pdf 2018.

[8] R. H. J. Giles, *Lasting Forests Glossary*, 2005.

[9] M. B. i. M. Markus Schatten, *Towards a General Definition of Biometric Systems*, vol. 2 ed. IJCSI International Journal of Computer Science Issues, 2009.

[10] What are Biometrics? Biometric Processes. , s Interneta, https://www.aware.com/what-are-biometrics-biometric-processes/ .

[11] National Institute of Standards and Technology - FIPS201 standard, note = , url = https://pages.nist.gov/FIPS201/introduction/, owner = agardijan, timestamp = 2005.

[12] What Is Biometric Authentication? A Complete Overview. , s Interneta, https://heimdalsecurity.com/blog/biometric-authentication/ 2021.

[13] Biometric Authentication, Identification and Verification in 2021. , s Interneta, https://recfaces.com/articles/biometric-authentication-identification 2020.

[14] D. A. T. i Bhaskar Krishnamachari, *Blockchain in a nutshell.* Springer Nature, 2022.

[15] N. R. K. S. Dylan Yaga, Peter Mell, *Blockchain Technology Overview.* National Institute of Standards and Technology Internal Report, 2019.06.26.

[16] Types of Blockchain: Public, Private, or Something in Between. , s Interneta, https://www.foley.com/en/insights/publications/2021/08/types-of-blockchain-public-private-between 2021.08.19.

[17] M. D. Maymounkov, P., *Kademlia: A peer-to-peer information system based on the xor metric.* Peer-to-Peer Systems, 2002.

[18] Z. C. Y. Q. Z. S. L. S. Wang, T., *Ethna: Analyzing the underlying peer-to-peer network of ethereum blockchain.* IEEE Trans. Netw. Sci., 2021.

[19] Dogecoin. , s Interneta, https://github.com/dogecoin/dogecoin/blob/master/README.md

[20] S. Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System.* www.bitcoin.org, 2008.

[21] H. M. M. P. Baird, L. Hedera: A public hashgraph network and governing council. , s Interneta, https://hedera.com/hh_whitepaper_v2.1-20200815.pdf 2020.

[22] C. A. K. M. L. E. G. A. Nguyen, Q., *Lachesis: Scalable asynchronous bft on dag stream.* Fantom, 2021.

[23] H. R. M. S. V. G. Z. N. Gilad, Y., *Algorand: Scaling byzantine agreements for cryptocurrencies. In: Proceedings of the 26th Symposium on Operating Systems Principles.* Association for Computing Machinery, New York, NY, USA, 2017.

[24] N. S. King, S. Ppcoin: Peer-to-peer crypto-currency with proof-of-stake. , s Interneta, https://www.peercoin.net/whitepapers/peercoin-paper.pdf 2012.

[25] N. Szabo. The idea of smart contract. , s Interneta, https://nakamotoinstitute.org/the-idea-of-smart-contracts 1997.

[26] Ethereum Virtual Machine Opcodes. , s Interneta, https://www.ethervm.io/

[27] ZK7: Miden VM: a STARK-friendly VM for blockchains - Bobbin Threadbare – Polygon. , s Interneta, https://www.youtube.com/watch?v=81UAaiIgIYA

[28] A. M. V. Cosimo Sguanci, Roberto Spatafora, *Layer 2 Blockchain Scaling: a Survey*, 2021.06.15.

[29] J. Benet, *IPFS - Content Addressed, Versioned, P2P File System*, 2014.07.14.

[30] IPFS: A Complete Analysis of The Distributed Web. , s Interneta, https://medium.com/zkcapital/ipfs-the-distributed-web-e21a5496d32d 2018.09.28.

[31] NFTs Explained: A Must-Read Guide to Everything Non-Fungible. , s Interneta, https://nftnow.com/guides/what-is-nft-meaning/ 2022.10.06.

[32] What are non-fungible tokens (NFTs)? , s Interneta, https://capital.com/non-fungible-tokens-nft-definition .

[33] R. T. i. R. V.-R. Oscar Delgado-Mohatar, Julian Fierrez, *Blockchain meets Biometrics: Concepts, Application to Template Protection, and Trends*. School of Engineering, Universidad Autonoma de Madrid, Madrid, Spain, 2020.

[34] J. J. Youn Kyu Lee, *Securing biometric authentication system using blockchain*. ScienceDirect, 2021.

[35] Polygon. Polygon ID. , s Interneta, https://polygon.technology/polygon-id 2023.

[36] ——. Core Concepts of Polygon ID: Verifiable Credentials, Identity Holder, Issuer and Verifier (Triangle of Trust). , s Interneta, https://0xpolygonid.github.io/tutorials/ 2023.

[37] i. M. K. Thomas Kerber, Aggelos Kiayias, *Kachina – Foundations of Private Smart Contracts*. University of Edinburgh and IOHK, 2021.

[38] ZKEVM OVERVIEW - What is a zkEVM? , s Interneta, https://www.alchemy.com/overviews/zkevm

[39] Ethereum EVM illustrated. , s Interneta, https://takenobu-hs.github.io/downloads/ethereum_evm_illustrated.pdf

[40] A. Yao., *How to generate and exchange secrets*, 1986th ed. In 27th FOCS, 162–167, 1986.

[41] S. M. i. A. W. O. Goldreich, *How to play any mental game – A completeness theorem for protocols with honest majority*, 1987th ed. In 19th STOC, 218–229, 1987.

[42] Y. Lindell, *Secure Multiparty Computation (MPC)*. Unbound Tech and Bar-Ilan University, 2020.

[43] A. Shamir, *SHow to Share a Secret*. CACM, 22(11):612–613, 1979.

[44] P. W. Shor, *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*, 1997th ed. SIAM Journal on Computing, 1997.10.01.

[45] D. Micciancio and O. Regev, *Lattice-based Cryptography*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 147–191. , s Interneta, https://doi.org/10.1007/978-3-540-88702-7_5

[46] A. S. Gillis. What is homomorphic encryption? . , s Interneta, https://www.techtarget.com/searchsecurity/definition/homomorphic-encryption 2022.

[47] Intel. Accelerating Fully Homomorphic Encryption with an Open Source FPGA Library . , s Interneta, https://www.intel.com/content/www/us/en/developer/articles/technical/homomorphic-encryption/accelerating-homomorphic-encryption-for-fpga.html 2022.

[48] What is Ethereum? , s Interneta, https://ethereum.org/en/what-is-ethereum/ .

[49] E. K. i Bruce Mizrach, *An Event Study of the Ethereum Transition to Proof-of-Stake*, 2022.10.24.

[50] Ethereum 2.0 depozitni ugovor. , s Interneta, https://etherscan.io/address/0x00000000219ab540356cBB839Cbe05303d7705Fa

[51] Ethereum Finality. , s Interneta, https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/finality

[52] Ethereum slashing. , s Interneta, https://beaconcha.in/validators/slashings

[53] Discover Cardano. , s Interneta, https://cardano.org/discover-cardano#research

[54] Ouroboros - An environmentally sustainable, verifiably secure proof-of-stake protocol with rigorous security guarantees. , s Interneta, https://cardano.org/ouroboros/

[55] A beginner's guide to the Cardano network and the ADA ecosystem. , s Interneta, https://cointelegraph.com/blockchain-for-beginners/a-beginners-guide-to-the-cardano-network-and-the-ada-ecosystem

[56] A. Y. Kazım Rıfat Özyılmaz, *Designing a blockchain-based IoT infrastructure with Ethereum, Swarm and LoRa.* IEEE Consumer Electronics Magazine, 2018.09.22.

[57] R. C. Merkle, *A digital signature based on a conventional encryption function.* Springer, 1988.

[58] C. Dannen, *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners.* Apress, 2017.

[59] IPFS command-line install. , s Interneta, https://docs.ipfs.tech/install/command-line/

[60] IPFS command-line quickstart. , s Interneta, https://docs.ipfs.tech/how-to/command-line-quick-start/#prerequisites

[61] IPFS command-line take your node online. , s Interneta, https://docs.ipfs.tech/how-to/command-line-quick-start/#take-your-node-online

[62] ArchWiki - Nix. , s Interneta, https://wiki.archlinux.org/title/Nix

[63] Archlinux.org - Nix package. , s Interneta, https://archlinux.org/packages/community/x86_64/nix/

[64] Installing the Cardano node. , s Interneta, https://docs.cardano.org/development-guidelines/installing-the-cardano-node

[65] Building Cardano Node with nix. , s Interneta, https://github.com/input-output-hk/cardano-node/blob/master/doc/getting-started/building-the-node-using-nix.md/

[66] How to run cardano-node. , s Interneta, https://developers.cardano.org/docs/get-started/running-cardano/#testnet--sandbox 2022.

[67] How to run cardano-node. , s Interneta, https://developers.cardano.org/docs/get-started/running-cardano/

[68] Pin files using IPFS - Three kinds of pins. , s Interneta, https://docs.ipfs.tech/how-to/pin-files/#three-kinds-of-pins .

[69] Pin files using IPFS. , s Interneta, https://docs.ipfs.tech/how-to/pin-files .

[70] NFT Maker. , s Interneta, https://www.nmkr.io/

[71] Pool PM NFTs. , s Interneta, https://pool.pm/nfts

[72] Cardano NFT Metadata Standard. , s Interneta, https://github.com/cardano-foundation/CIPs/pull/85

[73] Cardano Testnet Faucet. , s Interneta, https://docs.cardano.org/cardano-testnet/tools/faucet

[74] Cardanoscan Preview testna mreža za korištenu adresu. , s Interneta, https://preview.cardanoscan.io/address/60ce4b2cf062c369e65d36ba4fe113bd8df6324bc02dba0ea3dc4179b1 2023.

[75] K. Loesing and S. J. M. i Roger Dingledine, *A Case Study on Measuring Statistical Data in the Tor Anonymity Network*. Springer, 2010.

[76] Polygon zkEVM. , s Interneta, https://polygon.technology/polygon-zkevm

Pojmovnik

B block size. 10

BASH Bourne Again Shell. 21

CID content identifier. 27

DAG directed acyclic graph. 7, 12

EVM Ethereum Virtual Machine. 41, 42

IOHK Input Output Hong Kong. 23

IPFS interplanetary filesystem. 12, 50, 51

MPC multiparty computation. 43, 44, 46, 50, 51

NFT Non Fungible Token. 13, 21

P2P peer-to-peer. 6, 9, 12

POS proof-of-stake. 17, 19

POW proof-of-work. 17

SHA256 secure hashing algorithm 256. 6

TB time block generation. 10, 11

TPB transactions per block. 10

TPS transcations per second. 10

TR relay time. 10, 11

zkEVM Zero-Knowledge Ethereum Virutal Machine. 41, 47

Sažetak

Ovaj rad razmatra decentralizirane biometrijske metode za autentifikaciju korisnika prilikom čega je očuvana privatnost. Razvijen je koncept riješenja za privatnu decentraliziranu autentifikaciju koristeći infrastrukturu *blockchaina* i interplanetarnog datotečnog sustava. Navedene tehnologije su obje prisutne prilikom izrade nezamjenjivih tokena kroz pohranu podataka i njen integritet. Infrastrukture sloja 2 kao i posredničkih mreža *blockchaina* pružaju dodatne prednosti u aspektima skalabilnosti i privatnosti.

Ključne riječi **— blockchain, NFT, decentralizacija, biometrija, autentifikacija, privatnost**

Abstract

This paper considers decentralized biometric methods for authentification of users while preserving privacy. A concept was developed for private decentralized authentification by utilizing the infrastructure of blockchain and interplanetary filesystems. The mentioned technologies are both present during the minting of non fungible tokens by means of storage and integrity preservation. Blockchain layer 2 infrastructure as well as sidechain solutions provide additional benefits in the aspects of scalability and privacy.

Keywords **—blockchain, NFT, decentralization, biometry, authentification, privacy**