

Дизајн и имплементација EIP-7702-компатибилног *Ethereum* вишепотписног дигиталног новчаника

Design and Implementation of an EIP-7702-Compatible Ethereum Multi-Signature Wallet

Александар Стојановић, Факултет техничких наука, Нови Сад

Студијски програм – РАЧУНАРСТВО И АУТОМАТИКА

Кратак садржај – У овом раду представљен је вишепотписни *Ethereum* новчаник који решава два ограничења стандардног налога у власништву корисника (EOA): одсуство вишепотписне ауторизације и потребу за две одвојене трансакције при депоновању ERC-20 токена. Предложено решење комбинује вишепотписни паметни уговор са EIP-7702 механизмом делегације кода, чиме корисник у једној трансакцији атомски одобрава и депонује средства. Рад укључује и демонстрацију система стварним трансакцијама на Sepolia тест мрежи, којима је потврђена уштеда гаса од приближно 38% у односу на класичан приступ.

Кључне речи: блокчејн, вишепотписни уговори, EIP-7702, апстракција налога

Abstract – This paper presents a multisig *Ethereum* wallet that addresses two limitations of the standard externally owned account (EOA): the lack of multisignature authorization and the need for two separate transactions when depositing ERC-20 tokens. The proposed solution combines a multisig smart contract with the EIP-7702 code delegation mechanism, allowing a user to atomically approve and deposit funds in a single transaction. The work includes a demonstration of the system with real transactions on the Sepolia test network, confirming a gas saving of approximately 38% compared to the classic approach.

Keyword: blockchain, multisig contracts, EIP-7702, account abstraction

НАПОМЕНА: Овај рад проистекао је из мастер рада чији ментор је био др Душан Гајић, ред. проф.

1. УВОД

Блокчејн технологија, уведена *Bitcoin* мрежом [1], омогућила је настанак дигиталне имовине без централизованог посредника. Стандардни *Ethereum* налог у власништву корисника (*Externally Owned Account*, EOA) ауторизује сваку трансакцију искључиво једним приватним кључем, без могућности дефинисања прага сагласности више страна [2]. Ово представља јединствену тачку отказа:

компромитовање једног кључа даје неограничен приступ целокупној имовини налога. Вишепотписни (*multisig*) протоколи решавају овај проблем захтевањем сагласности M од N потписника пре извршења трансакције, на начин аналоган потписивању корпоративних рачуна у традиционалним финансијама.

Додатни проблем јавља се при раду са ERC-20 токенима [3]: стандард захтева да корисник пошаље трансакцију *approve()*, а тек затим паметни уговор може позвати *transferFrom()* у другој, засебној трансакцији. Корисник плаћа накнаду за гас два пута и потврђује две одвојене блокчејн операције, што је посебно изражено код вишепотписних новчаника, где више корисника дели власништво над заједничким средствима.

Овај рад представља вишепотписни *Ethereum* новчаник који комбинује M -од- N паметан уговор са механизмом EIP-7702 [4], уведеним *Ethereum* надоградњом *Pectra* [5], који омогућава да EOA привремено изврши код паметног уговора у оквиру једне трансакције. Систем је реализован као екстензија за *Chromium*-базиране претраживаче и тестиран на Sepolia тестној мрежи.

2. ВИШЕПОТПИСНИ ПРОТОКОЛИ И EIP-7702

Ethereum разликује два типа налога: EOA, контролисан приватним кључем и налог паметног уговора, чији је код трајно записан на блокчејну и извршава се на *Ethereum* виртуелној машини [6]. Вишепотписни уговор имплементира образац M -од- N тако што чува листу овлашћених потписника и праг потребних потписа, а свака трансакција пролази кроз три фазе: предлагање (*propose*), потписивање (*sign*) и извршавање (*execute*), аналогно протоколима попут *Safe* [7].

EIP-7702 [4] уводи трансакцију типа 4 која носи листу овлашћења (*authorization list*). Свако овлашћење садржи идентификатор мреже, циљну адресу паметног уговора и *nonce* вредност и потписано је приватним кључем налога који делегира извршавање. Када мрежа прихвати овлашћење, у код слот EOA налога уписује се делегацијски означивач – 23-бајтна вредност *0xef010* надовезана 20-бајтном адресом циљног уговора. Бајт *0xef* је резервисан као неважећи EVM бајткод још ранијим EIP-3541, чиме је

загарантовано да се означивач никада не може помешати са бајткодом правог паметног уговора. Од тренутка уписа, сваки позив ка том *EOA* налогу извршава код циљног уговора у контексту налога – *address(this)* унутар тог кода враћа адресу *EOA*-а, а не циљног уговора – све док се делегација експлицитно не опозове слањем овлашћења са нултом адресом. За разлику од потпуне апстракције налога какву предлаже *EIP-4337* [8], *EIP-7702* не захтева трајну промену типа налога, па налог остаје *EOA* пре, током и после делегације.

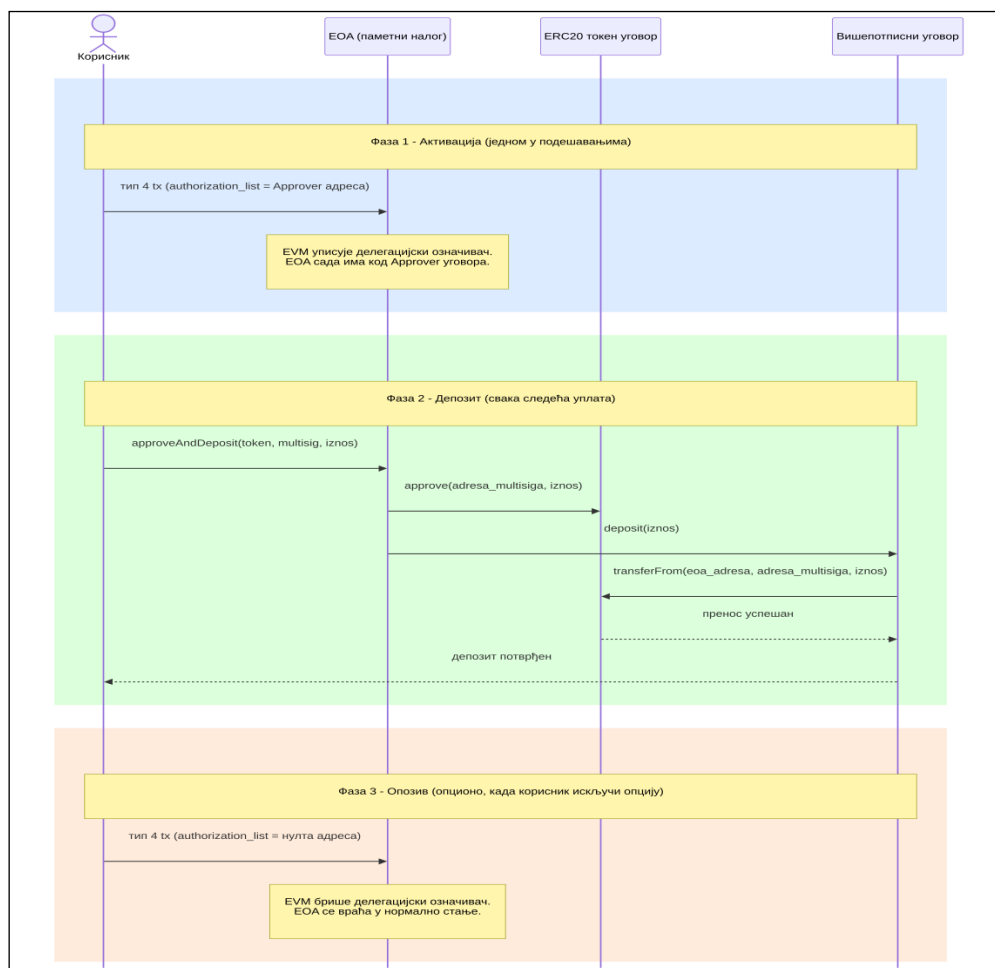
Практична употреба овог механизма за депоновање *ERC-20* токена одвија се у три фазе. У првој фази корисник једном активира „паметни налог“ слањем трансакције типа 4 чије овлашћење показује на *Approver* уговор. У другој фази, свако наредно депоновање одвија се као једна обична трансакција у којој *EVM* атомски извршава и одобрење и пренос средстава, јер *EOA* већ носи код *Approver* уговора. У трећој, опционалној фази, корисник опозива делегацију слањем још једне трансакције типа 4 чији циљ је нулта адреса, чиме се *EOA* враћа у првобитно стање обичног налога без кода.

2.1. Безбедносна својства

Свако овлашћење мора носити *nonce* вредност која одговара тренутној *nonce* вредности потписничког *EOA*-а, чиме се спречава напад поновном употребом потписа – једном искоришћено овлашћење не може се поново применити јер ће *nonce* у међувремену бити повећан. Идентификатор мреже у овлашћењу везује делегацију за конкретну *Ethereum* мрежу, па овлашћење потписано за *Sepolia* тестну мрежу не може бити преузето и примењено на главној мрежи. Ова два механизма заједно осигуравају да власник *EOA*-а у сваком тренутку има пуну контролу над тиме када, где и на који уговор је активан делегацијски означивач.

2.2. Историјат

EIP-7702 није први предлог овакве функционалности. Претходни предлог, *EIP-3074* [9], увео би два нова *EVM* оп-кода, *AUTH* и *AUTHCALL*, којима би паметан уговор могао да извршава позиве у име *EOA*-а који га је овластио, на сличан начин. Предлог је напуштен у корист *EIP-7702*, првенствено зато што увођење потпуно нових оп-кодова захтева трајне измене *EVM*-а, док *EIP-7702* исти циљ постиже увођењем новог типа трансакције, што је једноставније за интеграцију у постојећи протокол и алатке.



Слика 1. Трофазни ток депоновања *ERC-20* токена путем *EIP-7702* делегације: активација „паметног налога“, атомски депозит и опционални опозив делегације

3. ПОСТОЈЕЋА РЕШЕЊА

Више постојећих решења делимично адресира проблем безбедног управљања заједничким дигиталним средствима, али свако од њих доноси компромисе релевантне за овај рад.

Gnosis Safe [7, 10] је најзаступљенији вишепотписни новчаник на *Ethereum* мрежи, али је заснован искључиво на паметним уговорима: корисник објављује нови уговор који постаје власник средстава, док оригинални *EOA* губи улогу примарне адресе. Ово подразумева миграцију имовине и губитак *EOA* идентитета.

EIP-4337 [8] уводи апстракцију налога путем алтернативног *tempool*-а и *bundler* чворова који групишу корисничке операције. Овим механизмом могуће је имплементирати вишепотписне шеме, али и он захтева напуштање *EOA* модела и зависност од инфраструктуре која није део језгра *Ethereum* протокола.

Permit2 [11], решење *Uniswap*-а [12], уместо појединачних одобрења користи једно максимално одобрење ка самом *Permit2* уговору, након чега се свако наредно одобрење потписује ван ланца. Овим се решава проблем вишеструких одобрења, али по цену безбедносног компромиса: компромитовање *Permit2* уговора или потписивање злоћудне поруке омогућава нападачу приступ свим токенима које је корисник икада одобрио.

Ниједно од наведених решења не комбинује одржавање *EOA* идентитета, *M*-од-*N* вишепотписну ауторизацију и атомски депозит *ERC-20* токена без трајних неограничених одобрења трећим странама – управо ту празнину попуњава решење развијено у овом раду.

4. ИМПЛЕМЕНТАЦИЈА

Систем се састоји од два паметна уговора и клијентске *React* екстензије која комуницира са *Ethereum* мрежом путем *ethers.js* библиотеке [13], без посредног серверског залеђа – екстензија комуницира искључиво директно са *RPC* чвором мреже. Дељена пословна логика (вишепотписна ауторизација, *EIP-7702* делегација) у потпуности се налази у паметним уговорима, док клијентска страна само конструише, потписује и шаље трансакције, као и читава стање уговора.

Уговор *MultiSig* чува мапирање овлашћених потписника и праг *minSignatures*. Предлог трансакције идентификован је вредношћу *txHash* = *keccak256(posiljalac, nonce++)*, намерно независном од садржаја саме трансакције, како би се избегле колизије између истоветних узастопних предлога. Функција *execute* преноси средства тек када број потписа достигне праг и уговор располаже довољним износом.

Уговор *Approver* имплементира функцију *approveAndDeposit*, која редом позива *approve()* над *ERC-20* токеном и *depositToken()* над *MultiSig* уговором. Када је налог корисника делегиран на овај уговор путем *EIP-7702*, позив *approveAndDeposit* на

сопствену адресу извршава обе операције у оквиру једне трансакције, јер се *address(this)* унутар делегираног кода разрешава на адресу самог *EOA* налога, а не на адресу уговора *Approver*.

Клијентска страна конструише и потписује трансакцију типа 4 преко *ethers.js* библиотеке [13] позивом *signer.authorize({address, nonce})*, који гради и потписује овлашћење, након чега се шаље као самопозив – поље *to* трансакције једнако је сопственој адреси пошиљаоца, јер се *authorization_list* обрађује независно од одређишта трансакције. Најосетљивији део имплементације је тачна *nonce* вредност овлашћења: протокол прихвата овлашћење само ако се поклапа са тренутном *nonce* вредношћу потписника у тренутку обраде, а та обрада се извршава тек пошто је слање саме трансакције већ повећало *nonce* пошиљаоца за један. Када су пошиљалац трансакције и потписник овлашћења иста адреса, као у овом самопослатом обрасцу, овлашћење мора носити вредност *tekuci_nonce + 1*, а не *tekuci_nonce* прочитан непосредно пре слања. Стварна трансакција забележена на *Sepolia* мрежи ово потврђује: трансакција носи *nonce=2*, док потписано овлашћење унутар ње носи вредност 3, уз *chainId=0xaa36a7* (*Sepolia*). Након потврђивања, читавање кода налога (*eth_getCode*) враћа тачно *0xef0100* надовезано адресом *Approver* уговора, чиме је потврђено да је делегација примењена на мрежи. Опозив делегације користи идентичан образац, са нултом адресом као циљем овлашћења.

Клијентски слој излаже два одвојена позива за депоновање токена, тако да корисник може бирати између њих. Функција *depositTokenWithDelegation* претпоставља да је делегација већ активна и позива *approveAndDeposit* на сопствену адресу, чиме се цео депозит обавља једном трансакцијом. Насупрот томе, пар функција *approveTokenClassic* и *depositTokenClassic* репродукује класичан двотрансакцијски ток – прва позива *approve()* директно на токен уговору, а друга тек након тога *depositToken()* на *MultiSig* уговор. Ова два пута у коду су оно што омогућава директно мерење разлике у потрошњи гаса приказано у наредном поглављу, јер оба пута завршавају у истом крајњем стању *MultiSig* уговора.

5. ДЕМОНСТРАЦИЈА И РЕЗУЛТАТИ

Функционалност система потврђена је сценариом са три овлашћена потписника и прагом 2 од 3, извршеним на *Sepolia* тест мрежи над наменски објављеним *MultiSig* уговором и помоћним *ERC-20* токеном. Први корисник предлаже и потписује пренос 50 токена другом кориснику, други корисник додаје други потпис, након чега оба корисника депонују потребна средства – сваки користећи сопствени ток, са и без активне *EIP-7702* делегације.

Обједињени приступ захтева једно потписивање уместо два и смањује потрошњу гаса за приближно 38%, уз идентичан крајњи ефекат – уговор *MultiSig* прима исти износ токена, спреман за даље извршавање већ потписаног предлога.

Цео сценарио прати исту логичку путању кроз уговор *MultiSig*, независно од тога да ли је активна *EIP-7702* делегација: предлагач позива *propose*, чиме се у уговору уписује нов унос идентификован вредношћу *txHash*, остали потписници позивају *sign* над истим идентификатором, а уговор интерно пребројава потписе по сваком предлогу. Тек када број потписа достигне праг, било који учесник може позвати *execute*, који преноси средства само ако уговор у међувремену располаже довољним депозитованим износом – чиме су предлагање и потписивање потпуно независни од тренутка када су средства стварно депонована.

6. ЗАКЉУЧАК

Рад је показао да комбинација вишепотписног паметног уговора и *EIP-7702* делегације кода значајно унапређује корисничко искуство при раду са *ERC-20* токенима, без потребе за потпуном миграцијом налога на новчанике имплементираних уз помоћ паметног уговора (*smart contract wallets*). Исти образац – обједињавање више узастопних позива у једну трансакцију путем привременог понашања паметног уговора – није ограничен на депозит токена у вишепотписни уговор, већ се може применити на било који ток који данас захтева више одвојених трансакција.

6.1. Ограничења

Тренутна имплементација тестирана је искључиво на *Sepolia* тест мрежи и није прошла независну безбедносну ревизију, што је предуслов за разматрање употребе на главној мрежи. Управљање приватним кључем ослања се на директан унос или генерисање кључа и шифровање лозинком, без подршке за стандардизоване мнемоничке фразе (*BIP-39*) или хардверске новчанике. Фокус рада је намерно ограничен на вишепотписни уговор и *EIP-7702* делегацију, док подршка за вишепотписне трансакције у нативној *ETH* валути остаје ван домена овог рада.

6.2. Будући рад

Пре разматрања употребе на главној мрежи неопходна је независна безбедносна ревизија уговора *MultiSig* и *Approver*, као и усклађивање управљања приватним кључем са уобичајеном праксом индустрије. Приказ историје трансакција путем јавног *API*-ја и подршка за више *Ethereum* мрежа истовремено представљају природно проширење корисничког интерфејса. Са аспекта саме *EIP-7702* делегације, примењени образац није ограничен на депозит *ERC-20* токена, већ би могао постати општи механизам за побољшање корисничког искуства у ширем скупу *Ethereum* апликација.

7. ЛИТЕРАТУРА

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system", 2008. URL: <https://bitcoin.org/bitcoin.pdf> (pristupljeno u julu 2026)

[2] A.M. Antonopoulos, G. Wood, "Mastering Ethereum: Building Smart Contracts and DApps", O'Reilly Media, 2018.

[3] F. Vogelsteller, V. Buterin, "EIP-20: Token Standard", 2015. URL: <https://eips.ethereum.org/EIPS/eip-20> (pristupljeno u julu 2026)

[4] V. Buterin et al., "EIP-7702: Set EOA account code", 2024. URL: <https://eips.ethereum.org/EIPS/eip-7702> (pristupljeno u julu 2026)

[5] T. Beiko, "EIP-7600: Hardfork Meta - Pectra", 2025. URL: <https://eips.ethereum.org/EIPS/eip-7600> (pristupljeno u julu 2026)

[6] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger", 2024. URL: <https://ethereum.github.io/yellowpaper/paper.pdf> (pristupljeno u julu 2026)

[7] Safe Ecosystem Foundation, "Safe: Smart account overview", 2023. URL: <https://docs.safe.global/advanced/smart-account-overview> (pristupljeno u julu 2026)

[8] V. Buterin et al., "EIP-4337: Account Abstraction Using Alt Mempool", 2021. URL: <https://eips.ethereum.org/EIPS/eip-4337> (pristupljeno u julu 2026)

[9] S. Wilson, M. Zoltu, "EIP-3074: AUTH and AUTHCALL opcodes", 2020. URL: <https://eips.ethereum.org/EIPS/eip-3074> (pristupljeno u julu 2026)

[10] M. di Angelo, G. Salzer, "Wallet Contracts on Ethereum - Identification, Types, Usage, and Profiles", arXiv preprint, 2020. URL: <https://arxiv.org/abs/2001.06909> (pristupljeno u julu 2026)

[11] Uniswap Labs, "Permit2: Next Generation Token Approvals", 2022. URL: <https://github.com/Uniswap/permit2> (pristupljeno u julu 2026)

[12] Uniswap Labs, "Uniswap v3 Core", 2021. URL: <https://uniswap.org/whitepaper-v3.pdf> (pristupljeno u julu 2026)

[13] R. Moore, "ethers.js - Complete Ethereum library for JavaScript", 2024. URL: <https://docs.ethers.org/v6/> (pristupljeno u julu 2026)

Кратка биографија:



Александар Стојановић рођен је у Новом Саду 2000. год. Основне академске студије на Факултету техничких наука из области Електротехнике и рачунарства завршио је 2023. год. Мастер рад на Факултету техничких наука из области Рачунарства и аутоматике одбранио је 2026. год.

Контакт: al.stojanovic0@gmail.com