Зборник радова Факултета техничких наука, НовиСад

УДК: 007:004.056
ДОИ: https://doi.org/10.24867/34BE18Mijailovic

Имплементација аутентификације применом FIDO2 стандарда

Implementation of Authentication Using the FIDO2 Standard

Давид Мијаиловић, *Факултет техничких наука, Нови Сад*

Студијски програм – РАЧУНАРСТВО И АУТОМАТИКА

Кратак садржај – *Рад се бави дизајном, имплементацијом и анализом веб апликације која користи FIDO2 стандард као замену за традиционалне лозинке. Кроз теоријски преглед, развој функционалног прототипа и симулацију напада, рад демонстрира практичну примену и безбедносну отпорност система аутентификације без лозинки, нудећи увид у његову супериорност у односу на системе засноване на лозинкама.*

Кључне речи (три до пет): *Аутентификација, FIDO2, WebAuthn, аутентификација без лозинке, криптографија јавног кључа*

Abstract – *This paper addresses the design, implementation, and analysis of a web application that utilizes the FIDO2 standard as a replacement for traditional passwords. Through a theoretical overview, prototype development, and attack simulations, the work demonstrates the practical application and security resilience of a passwordless authentication system, offering insight into its superiority over password-based systems.*

Keywords: (three to five): *Authentication, FIDO2, WebAuthn, Passwordless Authentication, Public key cryptography*

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

1. УВОД

Дигитална трансформација је у средиште безбедносних изазова поставила концепт дигиталног идентитета, чија је верификација деценијама била готово синоним за употребу лозинки. Иако једноставне за имплементацију, лозинке су се показале као системски несигурне у модерном окружењу претњи. Истраживања су показала да корисници, оптерећени захтевима за дугим и сложеним лозинкама, усвајају небезбедне стратегије, као што је поновна употреба истих или сличних лозинки на више различитих сервиса [1]. Ова пракса, позната као „повезана судбина креденцијала“, значи да безбедносни пропуст на једном, мање битном сајту директно угрожава безбедност корисника на критичним сервисима попут имејла или интернет банкарства. Нападачи ово системски искоришћавају кроз нападе “попуњавањем креденцијала” (енгл. *credential stuffing*), где аутоматизовано тестирају процурене парове корисничких имена и лозинки на хиљадама других сајтова [2].

Поред напада на сервере, корисници су директно изложени нападима социјалног инжењеринга, пре свега „пецању“ (енгл. *phishing*). *Phishing* напади манипулишу корисницима како би их навели да добровољно унесу своје креденцијале на лажне веб странице које визуелно имитирају легитимне, што представља један од најраспрострањенијих и најефикаснијих проблема електронске крађе идентитета [3].

Као одговор на ове системске слабости, *FIDO* Алијанса (*Fast Identity Online Alliance*) и Конзорцијум за светску мрежу (*W3C*) развили су скуп стандарда познат као *FIDO2*. *FIDO2*, који се састоји од *Web Authentication* (*WebAuthn*) *API*-ја и *Client to Authenticator Protocol*-а (*CTAP2*), уводи отворену, скалабилну и интероперабилну архитектуру за аутентификацију без лозинки [4]. Коришћењем асиметричне криптографије, *WebAuthn* омогућава корисницима да се пријаве на онлајн сервисе користећи биометријске податке, мобилне уређаје или хардверске сигурносне кључеве на начин који је фундаментално отпоран на *phishing* и цурење база података са сервера [5].

Мотивација за овај рад произилази из препознавања да су инкрементална побољшања система заснованих на лозинкама достигла своје границе. Потребна је фундаментална промена парадигме. Примарни циљ овог рада је дизајнирати, имплементирати и анализирати функционални прототип веб апликације који у потпуности замењује традиционалну аутентификацију са *WebAuthn* стандардом, како би се демонстрирала његова практична применљивост и безбедносна отпорност.

2. НЕДОСТАЦИ ТРАДИЦИОНАЛНЕ АУТЕНТИФИКАЦИЈЕ

Традиционална аутентификација заснована на лозинкама рањива је на различите врсте напада који произилазе из комбинације техничких недостатака и

лоших навика корисника [6]. Упркос повећаној свести о важности сложености лозинки, многи корисници и даље примењују несигурне технике, као што су поновна употреба лозинки, одабир слабих комбинација и њихово записивање [6].

2.1. Phishing напади

Једна од најефикаснијих метода за крађу лозинки је *phishing* [7]. Нападачи креирају веб-сајтове који су готово идентични легитимним, а преваре су толико уверљиве да чак и искусни корисници могу бити преварени [8]. Основна рањивост лозинки на *phishing* произилази из њихове природе као дељене тајне.

2.2. Напади понављањем

Напад понављањем (енгл. *replay attack*) представља облик мрежног напада у којем се ваљан пренос података злонамерно понавља или одлаже [8]. Нападач може пасивно прислушкивати комуникацију, пресрести креденцијале за пријаву, а затим их поново послати да би остварио неовлашћен приступ. Да би се спречили овакви напади, неопходно је обезбедити „свежину“ сваке трансакције, што се постиже употребом механизама као што су једнократне лозинке или временске ознаке [9].

2.3. Man-in-the-Middle напади

Напад посредника (енгл. *Man-in-the-Middle*) је врста напада у којој нападач тајно пресреће и мења комуникацију између две стране које верују да комуницирају директно. У контексту лозинки, циљ је пресрести акредитиве током њиховог преноса. Иако се протоколи попут *HTTPS*-а користе за шифровање, нападачи и даље могу искористити рањивости [10]. Када се лозинка пресретне, она остаје важећа за пријаву и може се поново користити [11].

3. FIDO2 И WEBAUTHN СТАНДАРД

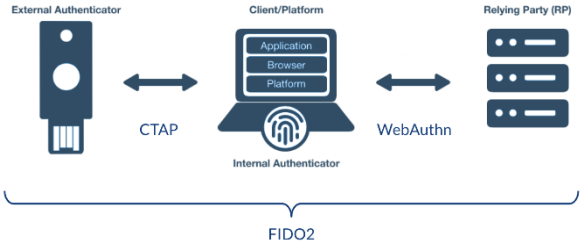
Стандард *FIDO2* представља комбинацију две међусобно повезане компоненте: *WebAuthn* и *CTAP2*. *WebAuthn* је спецификација коју је развио *W3C* и дефинише начин на који веб-апликације комуницирају са клијентским окружењем (прегледачем). *CTAP2* је спецификација под управом *FIDO* Алијансе и одређује како клијент комуницира са аутентификатором (нпр. *USB* токен, сигурносни кључ) [12].

Архитектура система укључује три кључна ентитета (слика 1) [12]:

1. *Relying Party* (*RP*) – Веб сервис (сервер) који жели да аутентификује корисника. *RP* иницира регистрацију, валидира криптографске потписе и чува јавне кључеве.

2. *WebAuthn* клијент / прегледач – Посредује између *RP*-а и аутентификатора користећи *JavaScript API* дефинисан у *WebAuthn* спецификацији.

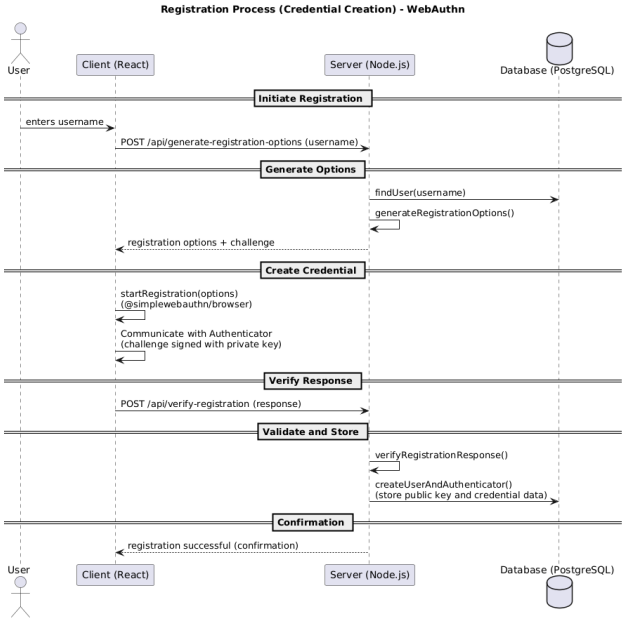
3. Аутентификатор – Компонента која безбедно чува приватни кључ и омогућава корисничку верификацију (нпр. биометрија, *PIN*). Може бити уграђен у уређај (*platform*) или бити спољни (*roaming*).



Слика 1. *Архитектура FIDO2 стандарда [13]*

4. ИМПЛЕМЕНТАЦИЈА И АНАЛИЗА СИСТЕМА

За потребе овог рада, развијен је функционални прототип веб апликације по клијент-сервер архитектури. Клијентска апликација је реализована као *Single Page Application* (*SPA*) коришћењем *React* библиотеке, док је серверска апликација изграђена на *Node.js* платформи уз *Express.js* окружење. За чување података коришћена је база података *PostgreSQL*. Архитектура система приказана је кроз секвенцијални дијаграм (слика 2) који илуструје ток комуникације између клијента, сервера и аутентификатора. На тај начин је обезбеђена јасна подела одговорности између слојева система и омогућена лакша надоградња и тестирање компонената.



Слика 1. *Секвенцијални дијаграм процеса регистрације корисника*

4.1. Клијентска и серверска имплементација

Клијентска апликација је реализована као *Single Page Application* коришћењем *React* библиотеке и *TypeScript*-а, при чему је кориснички интерфејс изграђен уз помоћ *Material-UI* компонентне библиотеке ради постизања визуелне конзистентности

и приступачности. Навигација унутар апликације имплементирана је помоћу *React Router*-а, који омогућава раздвајање јавних и заштићених рута [14]. Централизовано управљање статусом аутентификације спроведено је помоћу *AuthProvider* модула заснованог на *React Context API*-ју, који врши проверу активне сесије и одржава пријављено стање корисника.

На клијентској страни, коришћена је *@simplewebauthn/browser* библиотека која олакшава интеракцију са нативним *WebAuthn API*-јем прегледача. Процес аутентификације или регистрације одвија се у три корака:

1. Добијање опција са сервера: Клијент прво шаље *POST* захтев серверу са корисничким именом. Сервер враћа *JSON* објекат са опцијама, укључујући и криптографски изазов (енгл. *challenge*).

2. Покретање WebAuthn церемоније: Добијене опције се прослеђују функцији *startRegistration()* или *startAuthentication()*. Ове функције позивају нативни *WebAuthn API* прегледача (*navigator.credentials.create()* или *get()*), који преузима интеракцију са корисником и аутентификатором (нпр. *Windows Hello*, *YubiKey*).

3. Слање одговора на верификацију: Резултат добијен од аутентификатора (који садржи потписани изазов и друге податке) шаље се назад серверу на верификациону руту.

На серверској страни, коришћена је *@simplewebauthn/server* библиотека за генерисање изазова и верификацију криптографских одговора. Приликом регистрације, функција *generateRegistrationOptions()* се позива са параметрима као што су *rpName* (име апликације) и *rpID* (домен апликације). Посебно су важне опције унутар *authenticatorSelection* објекта: *userVerification: 'preferred'*, која сигнализира да треба захтевати верификацију корисника (PIN, биометрија), и *residentKey: 'required'*, која захтева од аутентификатора да сачува кључ на самом уређају, што омогућава аутентификацију без претходног уноса корисничког имена (*passkey*).

Функције *verifyRegistrationResponse* и *verifyAuthenticationResponse* врше криптографску проверу одговора добијеног од клијента. Оне упоређују изазов сачуван у сесији, проверавају да ли се *origin* и *rpID* поклапају са очекиваним вредностима дефинисаним у конфигурацији, и валидирају потпис користећи сачувани јавни кључ. За аутентификацију, функција такође проверава да ли је бројач (енгл. *counter*) у одговору већи од последњег сачуваног бројача у бази, што је кључна мера против клонирања аутентификатора.

Middleware ланац на серверу интегрише заштитне механизме као што су *rate limiting* и сесијско управљање изазовима, чиме се спречавају *brute-force* и *replay* напади. Логика је модуларно организована у контролере, руте и сервисе, што омогућава лако тестирање и одржавање кода.

4.2. Емпиријска анализа отпорности

Да би се практично потврдиле теоријске безбедносне гаранције, извршена је симулација три кључна вектора напада на имплементирани систем. Сваки напад је симулиран у контролисаном окружењу, користећи реалне услове комуникације између клијента и сервера. Резултати показују да комбинација механизама као што су *origin-binding*, јединствени криптографски изазов и бројач потписа обезбеђује вишеслојну заштиту против најчешћих типова напада.

Отпорност на Phishing
Симулиран је *phishing* напад подешавањем идентичне копије клијентске апликације на другом домену (порту). Покушај пријаве са лажне странице је био неуспешан. Заштита функционише на два нивоа [15]:

1. *CORS* политика: Серверска конфигурација је спречила комуникацију са неовлашћеног домена.

2. *Origin-binding*: Чак и након привременог онемогућавања *CORS* заштите, напад је заустављен на нивоу протокола. Потписани одговор који је аутентификатор генерисао на *phishing* сајту садржао је *origin* лажног сајта. Када је сервер верификовао овај одговор, одбио је аутентификацију јер се *origin* из потписа није поклапао са очекиваним, легитимним *origin*-ом апликације. Ово доказује да је заштита уграђена у технологију и не зависи од пажње корисника.

Отпорност на *Replay* нападе
Симулација је изведена пресретањем валидног аутентификационог одговора и његовим поновним слањем. Напад је био неуспешан захваљујући комбинацији два механизма:

1. Једнократни изазов: Сервер за сваки покушај пријаве генерише нови, јединствени изазов који се чува у сесији. Потписани одговор је валидан само за тај изазов.
2. Бројач потписа: Сваки *FIDO2* аутентификатор одржава интерни бројач који се инкрементира при свакој употреби. Сервер проверава да ли је бројач у примљеном одговору строго већи од последњег забележеног. Пошто поновљени захтев садржи исту вредност бројача, сервер га одбија [16].

Отпорност на *Man-in-the-Middle* нападе
WebAuthn архитектура је суштински отпорна на *MITM* нападе. Прво, спецификација захтева коришћење *HTTPS*-а, чиме се целокупна комуникација енкриптује. Друго, чак и у хипотетичком сценарију где нападач

успе да дешифрује саобраћај, напад је обесмишљен. За разлику од лозинке, у *WebAuthn* протоколу се не преносе осетљиви подаци које би нападач могао искористити. Приватни кључ никада не напушта сигурно окружење аутентификатора. Пресретнути потписани одговор је неупотребљив јер је једнократан и криптографски везан за оригиналну сесију и домен. Тиме је напад суштински обесмишљен јер нема тајне која се може украсти [17].

5. ЗАКЉУЧАК

Овај рад је демонстрирао дизајн, имплементацију и безбедносну ефикасност система аутентификације без лозинки заснованог на *FIDO2* стандарду. Кроз развој функционалног прототипа и емпиријску анализу, недвосмислено је потврђено да је имплементирано решење отпорно на кључне векторе напада који представљају системску претњу за традиционалне системе базиране на лозинкама, као што су *phishing*, *replay* и *Man-in-the-Middle* напади.

Показано је да је решење отпорно на *phishing* захваљујући криптографском везивању креденцијала за веб домен (енгл. *origin-binding*), што спречава злоупотребу на лажним сајтовима. Отпорност на *replay* нападе је осигурана комбинацијом једнократних криптографских изазова и бројача потписа. Коначно, анализа је показала да су *MITM* напади значајно отежани, јер приватни кључ никада не напушта сигурно окружење корисниковог уређаја. Резултати овог рада јасно потврђују да *WebAuthn* представља значајан искорак у дигиталној аутентификацији.

ЛИТЕРАТУРА

[1] D. R. Pilar, A. Jaeger, C. F. Gomes и L. M. Stein, „Passwords usage and human memory limitations: A survey across age and educational background," PloS one, т. 7, бр. 12, p. e51067, 2012.

[2] A. Das, J. Bonneau, M. Caesar, N. Borisov и X. Wang, „The tangled web of password reuse," у NDSS, 2014.

[3] M. Jakobsson и S. Myers, Phishing and countermeasures: understanding the increasing problem of electronic identity theft, John Wiley & Sons, 2007.

[4] FIDO Alliance, „FIDO Client to Authenticator Protocol (CTAP)," 2019.

[5] M. Weir, S. Aggarwal, B. De Medeiros и B. Glodek, „Password cracking using probabilistic context-free grammars," у 2009 30th IEEE Symposium on Security and Privacy, 2009.

[6] K. Chanda, „Password security: an analysis of password strengths and vulnerabilities," *International Journal of Computer Network and Information Security,* 2016.

[7] J. Owens и J. Matthews, „A study of passwords and methods used in brute-force SSH attacks," у USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET), 2008.

[8] RocketMeUpCybersecurity, „How WebAuthn is Changing the Future of Passwordless Security," 21 September 2023. Available: https://medium.com/@RocketMeUpCybersecurity/how-webauthn-is-changing-the-future-of-passwordless-security-71f71185fa42 (приступљено октобар 2025.)

[9] Y. Mo и B. Sinopoli, „Secure control against replay attacks," у 2009 47th annual Allerton conference on communication, control, and computing (Allerton), 2009.

[10] A. Mallik, „Man-in-the-middle-attack: Understanding in simple words," Cyberspace: Jurnal Pendidikan Teknologi Informasi, 2018.

[11] M. Conti, N. Dragoni и V. Lesyk, „A survey of man in the middle attacks," IEEE communications surveys & tutorials, 2016.

[12] M. Barbosa, A. Boldyreva, S. Chen, K. Cheng и L. Esquível, „Privacy and Security of FIDO2 Revisited," Proceedings on Privacy Enhancing Technologies, 2025.

[13] A. Sinitsyna, „Beyond Passwords: FIDO2 AND WebAuthn in practice,". Available: https://www.inovex.de/de/blog/fido2-webauthn-in-practice/ (приступљено октобар 2025.)

[14] Remix, „React Router," 2025. Available: https://reactrouter.com/home. (приступљено октобар 2025.)

[15] L. S. Huang, Z. Weinberg, C. Evans и C. Jackson, „Protecting browsers from cross-origin CSS attacks," у Proceedings of the 17th ACM conference on Computer and communications security, 2010.

[16] G. Dua, N. Gautam, D. Sharma и A. Arora, „Replay attack prevention in Kerberos authentication protocol using triple password", 2013.

[17] M. Al-Sinani и A. A. Zaidan, „A review on man-in-the-middle attacks in cloud computing and their detection and prevention," ACM Computing Surveys (CSUR), 2021.

Кратка биографија:

Давид Мијаиловић рођен је 2000. године у Лозници. Основне академске студије је завршио 2023. године на Факултету техничких наука у Новом Саду. **Контакт**: mijailovicd00@gmail.com