International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 05 | May - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Fast Identity Online 2 :Authentication Technique

Bhavana R M[1], Dr. Ganavi M [2]

[1] VIII Sem student, Department of CSE, JNN College of Engineering

[2] Assistant Professor, Department of CSE, JNN College of Engineering

1. rmbhavana2003@gmail.com 2. gaanavi4@jnnce.ac.in

Corresponding Author: rmbhavana2003@gmail.com

Abstract

Fast Identity Online 2 (FIDO2) emerges as a transformative solution to the vulnerabilities inherent in traditional password-based authentication methods. Leveraging public key cryptography and authenticators, it establishes a passwordless authentication paradigm, extending its relevance beyond web applications to diverse realms such as online payments and government services. This paper explores FIDO2's emphasis on a seamless user experience while bolstering security measures through innovative credential management techniques. The acceptance of FIDO2 on major browsers ensures its usability on mobile devices, with most modern devices equipped to support FIDO2 authentication, thus expanding its reach and applicability. Additionally, its adoption by major tech companies and standards bodies underscores its credibility and potential for widespread adoption. However, challenges remain, including overcoming legacy systems, addressing compatibility issues, and ensuring user education. Despite these challenges, FIDO2 represents a significant advancement in online authentication, offering strong security, usability, and privacy features, positioning it as a key enabler of the passwordless authentication paradigm.

Keywords: Public key cryptosystem, challenge, relying party, web browser and web authentication.

1. INTRODUCTION

In the ever-evolving landscape of cybersecurity, robust authentication mechanisms are increasingly essential. Traditional password-based methods, once central to online security, are vulnerable to various threats, from phishing to brute-force attacks. In response, FIDO2 emerges as a significant evolution, leveraging public key cryptography and authenticators to redefine authentication [1].

FIDO2 introduces a framework utilizing public key cryptography, where entities prove their identity through cryptographic keys: a public key, shared openly, and a private key, kept secret. Authentication involves the verifier sending a challenge, which the prover signs with its private key, and the verifier verifies the signature using the prover's public key, ensuring strong security, non-repudiation, and scalability. FIDO2 comprises two key protocols: WebAuthn and CTAP2 as shown in Fig 1. WebAuthn simplifies registration and authentication using authenticators, enhancing security by eliminating passwords. CTAP2 facilitates communication between the authenticator and applications or operating systems, ensuring seamless integration and interoperability. FIDO2 emphasizes "something we are" or "something we possess" over the traditional "something we know" approach, utilizing authenticators and biometrics for authentication. This shift introduces a new era of authentication based on inherent traits or possession

of secure hardware tokens. This paper explores FIDO2's principles, protocols, and transformative impact on online security [2].

Fig 1. Overview of FIDO2

2. WORKING OF FIDO2

FIDO2 combines WebAuthn and CTAP2 for seamless, passwordless authentication. WebAuthn handles registration and authentication for web applications, eliminating passwords for improved user experience. CTAP2 acts as a secure communication protocol between the client and authenticator, enabling tasks like registration and authentication. Together, they offer a flexible and interoperable framework for secure authentication, revolutionizing online security while prioritizing both security and usability.

2.1 Web Authentication API (WebAuthn)

Web Authentication (WebAuthn) is an API developed by the FIDO Alliance and W3C for accessing credentials in FIDO2. It enables passwordless registration and authentication to a web app, known as a relying party (RP), using authenticators. Authenticators can be platform-based, like laptops or smartphones, or roaming, such as the Yubico YubiKey security key. They store cryptographic keys necessary for WebAuthn ceremonies, enhancing security by resisting phishing attempts. WebAuthn involves two types of operations: "webauthn.create" for registration and "webauthn.get" for authentication. During these ceremonies, the RP sends a cryptographic challenge to the client, which communicates with the authenticator via CTAP2. For registration, the authenticator generates a new cryptographic key pair and stores the private key. For authentication, it creates a digital signature with the previously generated private key. The resulting object is sent back to the client and then to the RP for verification. Post-registration, the RP stores the username, public key, and other details for future authentication attempts [3].

2.1.1 Registration in WebAuthn

In a WebAuthn registration process as shown in Fig 2, the user begins by entering a username on a FIDO2-enabled form. A browser popup prompts the user to pair their smartphone with their laptop via Bluetooth. Upon successful pairing, the smartphone, equipped with touchID, verifies the user's identity. The smartphone acts as the authenticator, confirming the user's identity. Finally, the browser indicates the completion of the registration process, ensuring a seamless and secure experience.

Steps of WebAuthn registration:

1. Client requests registration from server.

2. Server responds with a JSON object, publicKeyCredentialCreationOptions, containing parameters set by the Relying Party (RP), including a randomly generated challenge and user information.

3. Client sends publicKeyCredentialCreationOptions to authenticator via CTAP2.

4. Authenticator generates a new key pair, potentially performing actions based on RP parameters.

5. Authenticator sends AttestationObject (containing authenticator data, attestation statement, and format) back to client.

6. Client creates clientDataJSON containing challenge, RP domain, and operation type, then sends AuthenticatorAttestationResponse to server.

7. RP processes and cross-checks information, stores public key, username, and data in a database.

During registration, parameters like attestation and key storage significantly influence trust. Attestation helps verify the authenticator's identity, providing crucial information to the RP. The WebAuthn API offers two methods for securely storing private keys: within the authenticator's memory or encrypted within the RP's database.

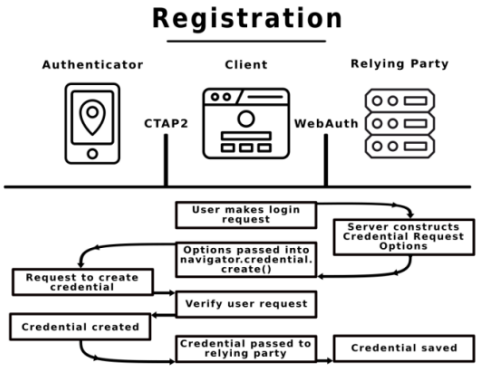

Fig 2. Registration process

2.1.2 Authentication in WebAuthn

In a hypothetical login scenario using FIDO2 as shown in Fig 3, a user utilizes a laptop browser and a smartphone as an authenticator. After entering the username, the browser prompts pairing with the smartphone via Bluetooth. A notification is sent to the smartphone, requesting confirmation of the login attempt via touchID. Upon consent, the browser displays a "successful authentication" message, granting access. This process eliminates the need for password recall, enhancing both security and user experience, showcasing FIDO2's seamless authentication with biometrics and device pairing replacing traditional passwords.

Authentication steps:

1. Client requests authentication.

2. RP sends authentication challenge (PublicKeyCredentialRequestOptions) to client.

3. Client sends hash of clientData and challenge to authenticator.

4. Authenticator verifies user and signs challenge with credential's private key.

5. Authenticator sends assertion signature, signCount, and userHandle to client.

6. Client constructs AuthenticatorAssertionResponse object with main fields.

7. Server validates assertion signature and received challenge, initiating session upon successful verification.

Important details for authentication protection include browser cross-checking domain origin hashes to prevent phishing attacks and using an incrementing counter to detect malicious imitations of authenticators [4].

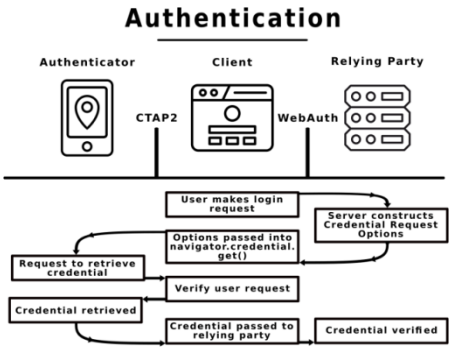

Fig 3. Authentication process

2.2 Client to Authenticator Protocol 2 (CTAP2)

CTAP2 facilitates communication between FIDO2-enabled clients and authenticators over BLE, NFC, or USB, providing an application layer protocol for interaction. The client initiates a connection and determines authenticator capabilities. Commands like authenticatorMakeCredential or authenticatorGetAssertion are sent to the authenticator, which responds with data or errors. CTAP2 comprises three parts: the Authenticator API, defining commands for communication; Message Encoding, specifying data encoding; and Transport Specific Bindings, detailing usage of USB, BLE, and NFC.

3. WEBAUTHENTICTIN ALGORITHMS

During FIDO2/WebAuthn registration, the cryptographic algorithm for generating a public key pair is determined by both the Authenticator's capabilities and the Relying Party's preferences, specified in the pubKeyCredParams of the PublicKeyCredentialCreationOptions JSON object.

To accommodate different transports like Bluetooth or NFC, data is translated into the Concise Binary Object Representation (CBOR) format due to potential bandwidth restrictions. The RP further refines supported algorithms based on those on the FIDO2 server. While authenticators have subsets of supported algorithms, during registration, the highest-priority algorithm set by the RP is chosen. The authenticatorGetInfo CTAP2 command offers insights into supported algorithms; for example, YubiKeys support RS256 (Firmware 5.1.X and below), EdDSA, ES256, and Ed25519 (Firmware 5.2.X and above). Commonly used algorithms in WebAuthn include ES256 (Elliptic Curve with SHA-256), RS256 (RSA with SHA-256), and EdDSA (Edwards-curve Digital Signature Algorithm), ensuring robust cryptographic protection for user authentication [5].

4. APPLICATIONS OF FIDO2

Some potential applications of FIDO2 are

1. Online Banking: FIDO2 enhances security and simplifies authentication for online banking platforms.

2. E-commerce & Payments: Improves security for e-commerce transactions and protects financial information.

3. Healthcare Systems: Secures access to electronic health records and telemedicine platforms.

4. Government Services: Enhances security for government portals, tax filing systems, and identity verification.

5. Enterprise Authentication: Strengthens authentication for employee access to corporate networks and applications.

6. Education Platforms: Ensures secure access to learning management systems and student portals.

7. Cross-Platform Authentication: Enables seamless and secure authentication across web browsers, mobile devices, and desktop applications.

5. CONCLUSION

FIDO2 revolutionizes online authentication by removing reliance on passwords, ensuring both security and usability. Its applications extend across sectors, from online payments to session management, addressing the need for stringent access control. Widely adopted by major tech players and supported by leading browsers like Apple and Chrome, FIDO2's prominence grows steadily. Challenges like user readiness and interoperability exist, but FIDO2's trajectory suggests a promising future. Its mainstream adoption reshapes authentication, promising safer and more user-friendly experiences. FIDO2's success paves the way for further innovation, ensuring security and usability advance together, heralding a new era in digital security.

REFERENCES

[1] Eleftherios Georgiadis, “FIDO2 Overview, Use Cases, and Security Considerations”, ResearchGate, pp. 11-46, Vol 1, 2023.

[2] Kepkowski, Michal and Machulak, Maciej and Wood, Ian and Kaafar, Dali, “Challenges with Passwordless FIDO2 in an Enterprise Setting: A Usability Study”, IEEE, pp. 37-48, Vol 2, 2023

[3] Grammatopoulos, Athanasios Vasileios and Politis, Ilias and Xenakis, Christos, “Blind software-assisted conformance and security assessment of FIDO2/WebAuthn implementations”, J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl , pp. 96-127, Vol 13, 2022.

[4] Bicakci, Kemal and Uzunay, Yusuf, “Is FIDO2 Passwordless Authentication a Hype or for Real?: A Position Paper”, IEEE, pp. 68-73, 2022.

[5] Zachary Breit, Hunter Dean, Tai-Juan Generrette, Samuel Howard, Balaji Kodali, Jim Kong, Jonah Tash, Phillip Wang, and John Wu, “Exploration of the Security and Usability of the FIDO2 Authentication Protocol” , IEEE, pp 27- 48, Vol. 2, 2022.