# Password-Less Authentication for Mobile Web Application

Ajay Balaji D

## Abstract:

Password authentication is standard and widely used because it is cost-effective and easy to manage. But it is challenging for the users in complying with strong password policies on choosing and keeping them private in heavily regulated industries. Recent studies highlighted that the password approach is responsible for 81% of hacking-related breaches either by week and/or stolen credentials. In addition to that, storing the millions of user credentials (including Plain / Encrypted passwords) in a cloud or an on-prem server can render a single point of the target for hackers. Passwords can grant access to sensitive data, it is critical to protect them or even better not to use them for login. As society becomes more dependent on the internet and web services, we need to find a replacement authentication method that users are willing to use. The newest contender for succeeding passwords as the incumbent web authentication scheme is the FIDO2 standard. In this work we are trying to implement FIDO2 standards for our use case with slight modification.

## 1. Introduction

Passwords are the most common form of online authentication, but each year the risk associated with passwords becomes more apparent. Expecting a user to remember a unique password for each account is impractical and has caused excessive password reuse. Password reuse by itself is not detrimental to overall account security, but because of other security vulnerabilities, such as data breaches and hacked accounts, password reuse is highly discouraged. The combination of data breaches and password reuse has caused many users to be susceptible to credential-stuffing attacks.

Based on free, open standards from the FIDO Alliance, Fast IDentity Online (FIDO) authentication enables password-only logins to be replaced with secure, fast login experiences across websites and apps. This is accomplished by using standard public-key cryptography to provide strong authentication and leave zero data at rest. FIDO U2F is an open standard that provides added security and simplifies Universal 2-Factor authentication.

During registration with an online service, the user's client device creates a new key pair. It retains the private key and registers the public key with the online service. Authentication is done by the client device proving possession of the private key to the service by signing a challenge. The client's private keys can be used only after they are unlocked locally on the device by the user. The local unlock is accomplished by a user inserting a FIDO2 Security Key or pressing the NFC button on the security key.

Most major browsers now support FIDO2, including Google Chrome, Edge, Firefox, and Safari. In FIDO2, authenticator has two types, cross-platform and platform-specific. Cross-platform authenticators are roaming devices, such as hardware tokens. Platform-specific authenticators only work for the device in which they reside, such as Face ID and Touch ID on Apple products or Windows Hello on Microsoft products.

The organization of this paper is as follows, Section 2 describes the related research and limitation of current method. In Section 3 we describe our hypothesis and calculated some metric to prove why our approach is better. We conclude in section 4 and discuss future work.

## 2. Related Research

The study, by Lyastani et al., involved ordinary users and was a lab study [1]. The users were given hardware tokens and completed tasks, which included registering and logging into websites provided by the researchers. After completing the tasks, users completed surveys to

measure their opinions about FIDO2 hardware tokens as a password replacement. According to the survey, one of the advantages of password-less authentication is that users no longer have to remember passwords. On the flip side, users now have to carry around a hardware token, and 39% of participants disliked the need to carry an additional item. Users also worried about losing access to their accounts if the hardware token was lost or destroyed. There were other concerns with the use of hardware tokens, such as new devices that lack USB ports, accessing an account on a public device, allowing relatives to login to their account, and revocation.

And also, we can't completely rely on biometric based authentication such as fingerprint readers, eye scanners, and facial recognition for websites [4]. Biometrics are considered acceptable for authentication for a personal device but not always for websites. One issue that applies to all forms of biometrics is that biometrics are not unique across websites. If someone uses their fingerprint to log into the website "A" and into website "B" and website "A" leaks the fingerprint, then the user is no longer secure on the website "B."

| Security | Password | Password-Less |
|---|---|---|
| Resilient-to-Physical-Observation | ✗ | ✓ |
| Resilient-to-Targeted-Impersonation | ✗ | ✓ |
| Resilient-to-Throttled-Guessing | ✗ | ✓ |
| Resilient-to-Unthrottled-Guessing | ✗ | ✓ |
| Resilient-to-Internal-Observation | ✗ | ✓ |
| Resilient-to-Leaks-from-Other-Verifiers | ✗ | ✓ |
| Resilient-to-Phishing | ✗ | ✓ |
| Resilient-to-Theft | ✓ | ✗ |
| No-trusted-Third Party | ✓ | ✓ |
| Requiring-Explicit-Consent | ✓ | ✓ |
| Un-linkable | ✓ | ✓ |

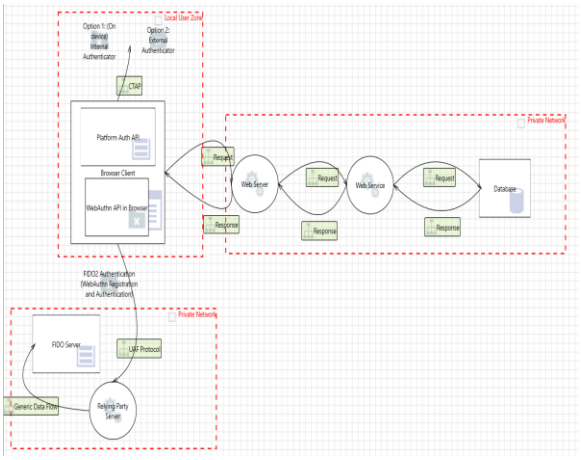
## 3. Hypothesis and Empirical Evidence

Our use case is cross platform-based password-less authentication using FIDO2 standards on mobile devices without hardware tokens. In our use case, the user will register their login device (Mobile In our case) with standard public-key cryptography technique so that their private key will not leave their device. Then at the time of authentication, to unlock the private key in the device we are going to use a software token instead of a hardware token. Our token will be generated in the same device where we are accessing the site, we no need to carry any extra device. This will eliminate the risk related to the hardware tokens and the risk of posting our biometric information to websites.

To achieve this, we need communication between the browser and the mobile device. Google is currently working on Cloud Assisted BLE (caBLE) which can be used in the future. Authenticator apps in mobile phones can be used to achieve this.

| Security | Software Token | Hardware Token |
|---|---|---|
| Hardware Free | ✓ | ✗ |
| Password-less Authentication | ✓ | ✓ |
| No USB compact worries | ✓ | ✗ |
| Reduced total cost of ownership | ✓ | ✗ |
| public-key cryptography usage | ✓ | ✓ |

Previous studies [1] [3] shows how the web application can be authenticated from laptop or desktop using hardware token and studies were unclear about accessing the website in the mobile device. Some studies shows if the website need to be accessed using mobile device then we need to use NFC as authenticator for FIDO Based approach. Now a days many people are using their mobile phones to access the websites and it will be hard for us to carry a hardware device to authenticate every time. Recent study shows Platform-specific authenticators are used in mobile devices to authenticate site in FIDO2 based approach. Our research goal is to generate the token in the same device from which the user is trying to access the site. This will have a limitation such that this approach can be suitable only for mobile devices for which we need a cross platform-based authentication.




Threats associated with FIDO authenticator vs Software Token:

| Threats | FIDO Authenticator | Software Token |
|---|---|---|
| Malicious Authenticator - Attacker convinces users to use a maliciously implemented authenticator. | Fail | Fail |
| U-auth private Key Compromise - Attacker succeeds in extracting a user's cryptographic authentication private key for use in a different context. | Fail | Pass |
| User Verification By-Pass - Attacker could use the cryptographic authentication key (inside the authenticator) either with or without being noticed by the legitimate user. | Fail | Pass |
| Physical Authenticator Attack - Attacker could get physical access to Authenticator (e.g., by stealing it). | Fail | Fail |

# 4. Conclusion and Future work

FIDO based authentication are popular and many top organizations have already adopted their approach but still lot of organizations are skeptical about adapting to their approach as there will be some cost involved when we are going to provide hardware authenticator for all the employees and it will not be possible when this method is implemented for a public facing site. If we started using software token instead of hardware authenticator then it would be easy for end users. Our use case is limited and If FIDO based authentication is possible without a need of hardware authenticator then everyone can try to implement it in public facing sites.

# 5. References

[1] Lyastani, S. G., Schilling, M., Neumayr, M., Backes, M., and Bugiel, S. "Is FIDO2 the kingslayer of user authentication? a comparative usability study of FIDO2 passwordless authentication". In IEEE Symposium on Security and Privacy (SP) (2020), IEEE, pp. 268-285.

[2] Michitomo Morii, Hiroki Tanioka, Kenji Ohira, Masahiko Sano, Yosuke Seki, Kenji Matsuura and Tetsushi Ueta. "Research on Integrated Authentication Using Passwordless Authentication Method". In IEEE 41st Annual Computer Software and Applications Conference (COMPSAC) (2017), IEEE, pp. 682-685.

[3] Fatima Alqubaisi, Ahmad Samer Wazan, Liza Ahmad, David W Chadwick. "Should We Rush to Implement Password-less Single Factor FIDO2 based Authentication?". In IEEE 12th Annual Undergraduate Research Conference on Applied Computing (URC) (2020), IEEE.

[4] Kim, H.-J. Biometrics, is it a viable proposition for identity authentication and access control? Computers & Security 14, 3 (1995), 205-214.