

How WebAuthn Works



okta



WebAuthn

In recent years, the cybersecurity industry has experienced a significant increase in websites losing consumer data to bad actors. Account takeover fraud losses have reached over \$5.1 billion, and leading customer facing companies like Marriott Starwood and Google have experienced massive data security breaches. With the fundamental loss in consumer trust and millions in lost revenue, companies must take a second look at their approach to protecting their users.

Index



- Part I: WebAuthn & Business Effects (Why You Care)** 4
- Part II: History of Authentication** 6
 - Shortcomings of Current WebAuthn Methods 6
 - Stakeholders of WebAuthn 7
- Part III: How WebAuthn Works & Okta UI** 9
 - Enrollment Flow 10
 - How Authentication Works 12
 - How Account Recovery Works 15
- Part IV: Moving Towards a Passwordless Future** 16

Part I:

WebAuthn & Business Effects

In March 2019, the World Wide Web Consortium (W3C) announced that WebAuthn is now the official web standard for password-free login. With support from a broad set of applications, widespread adoption of WebAuthn is expected in coming years.

With this new standard, any web application running in a browser that supports WebAuthn can now take advantage of these authenticators to securely authenticate users. Google Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari, Opera—and rapid adoption by platforms (e.g. MSFT Edge with Windows Hello, Google Android), means there's finally a practical way to deploy FIDO2/WebAuthn. In the past, there was a lack of supported browsers and platforms which was a huge impediment to the adoption of U2F. Now, organizations can deploy with external tokens like YubiKeys or through supported platforms themselves since the device itself is used for WebAuthn.



IT & Consumer benefits

WebAuthn has many positive side effects for both users and businesses. Not only are there better user experiences possible with WebAuthn (passwordless login, for example), but businesses and consumers alike benefit from better security and a decrease in the type of attacks that can be used against them.

For businesses, using WebAuthn means that automated, cross-site password attacks like password spraying or password stuffing can't be used against their websites. This also benefits users since it's usually their login credentials that are being leveraged for these attacks. In Part II, we will discuss which main stakeholders WebAuthn benefits.

For customers and end users, the major benefit is the prevention of account takeovers from phished credentials. Since WebAuthn works by registering the authenticator with a specific application, creating a public-private key pair, and storing credentials locally on built-in authenticators (like Microsoft Edge/Android) or external ones like YubiKeys, these credentials can't be easily phished.

WebAuthn will allow end users to take security into their own hands instead of relying completely on any specific organization to store their credentials properly or hope it won't be breached. And since WebAuthn is now a standard that is relatively easy to use, there's a greater chance that more ordinary people will use this method of strong authentication.

History of Authentication

Shortcomings of Current Authentication Methods

Before we get into why WebAuthn was created and its diverse functionality, we first have to understand how authentication of users started, and shortcomings of current methods.



Password Credentials

We are all too familiar with using usernames and passwords as a method for authentication. Although this framework is ubiquitous and easy to understand for the common consumer, [1 in 5 Americans](#) have experienced account takeover (ATO) from compromised password credentials. With the average end user having over [130 online accounts](#), businesses should look to mature their framework of user authentication for the benefits it will have on their business.

Two-Factor Authentication

The next iteration of credential authentication is 2FA, or two-factor authentication. However, for customer accounts, popular low assurance second factors like SMS can be subject to phishing attacks that have been [well documented](#). With the fallibility of 2FA, WebAuthn provides a potential solution to advanced phishing attacks while simultaneously improving customer experience.

What is WebAuthn?

Meet the new global standard of web authentication. WebAuthn is a browser-based API that allows for web applications to simplify and secure user authentication by using registered devices (phones, laptops, etc) as factors. It uses [public key cryptography](#) to protect users from advanced phishing attacks.

WebAuthn Benefits & Stakeholders

The advantages of using WebAuthn can be categorized into 3 main groupings affecting a variety of stakeholders including: customers, product owners, security teams, and support teams.

Benefits	Stakeholders
Enhance customer engagement	Customers, Product Owners
Strengthen security posture	Customers, Security teams
Reduce organizational load to support app <ul style="list-style-type: none">• WebAuthn as a primary method of login• Backwards Compatibility	Support

Enhance Customer Engagement

Customers—Frictionless Login Experience

Since WebAuthn uses device-based authentication, this wholly removes the need for passwords. For the customer, this means not having to remember your username and password when logging in, or trying to get a one-time password for a step up second factor. The authentication flow is simplified for the end user to just using their registered device for authentication.

Product Owners—Time to Authentication

As mentioned before, WebAuthn removes the need for passwords. Product owners care about the use of their application, and removing any barriers to customers is usually their primary goal. WebAuthn helps creating a frictionless login experience.

Product Owners—Time to Market

In addition, since WebAuthn removes the need to think about complex password settings, product owners can accelerate their time to market by avoiding the need to build complex architectures to manage and store passwords.

Strengthen Security Posture

Customers—Preservation of Trust

With the proliferation of data breaches, consumer trust is especially important to maintain. Customers are giving you their information, and they want to know their data is safe when sharing it. With WebAuthn, you are getting a much more secure authentication method that subverts the [risks of passwords](#).

Security Teams—Eliminates Inherent Weakness of Passwords

WebAuthn does not rely on knowledge based authentication such as usernames and passwords. This means that you are relying on registered devices that belong to the end user. The risk of spoofing authentication is lower because registered physical devices are harder to steal than passwords, making your security team happier.

Reduce Organizational Load to Support Applications

Support Organization—Reduce Support Cycles for Multiple Factors

A unique functionality of WebAuthn is using this standard as the primary method of logging in. By implementing WebAuthn, support cycles will be eased since the number of factors to enroll for will essentially be reduced to just WebAuthn.

Support Organization—Backwards Compatibility

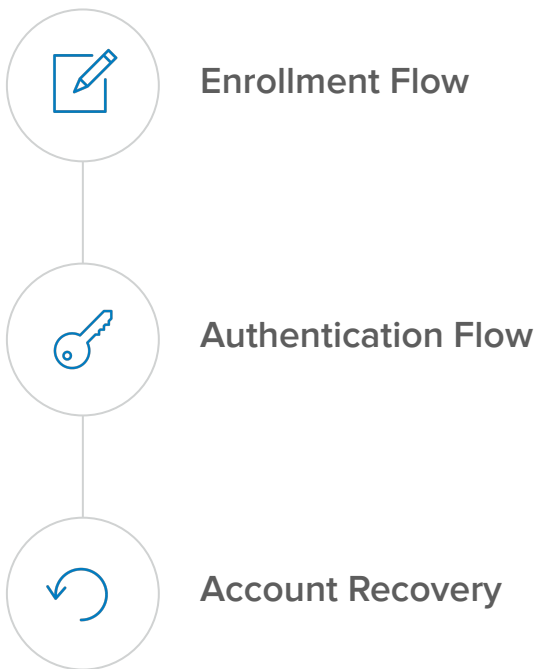
In addition, WebAuthn includes backwards compatibility which means that organizations don't need to refresh tokens immediately that have already been distributed. FIDO U2F keys like U2F supported YubiKeys still work as second factors with web browsers that support WebAuthn. This means that organizations can have slower or more planned migration.

Part III:

How Does WebAuthn Work? (Flows & Okta UI)

We've walked through some examples of how WebAuthn can benefit your customer experience as well as strengthen your security posture. Now, we'll take a look under the hood on the flows of WebAuthn, and best practices associated with each part of the WebAuthn flow.

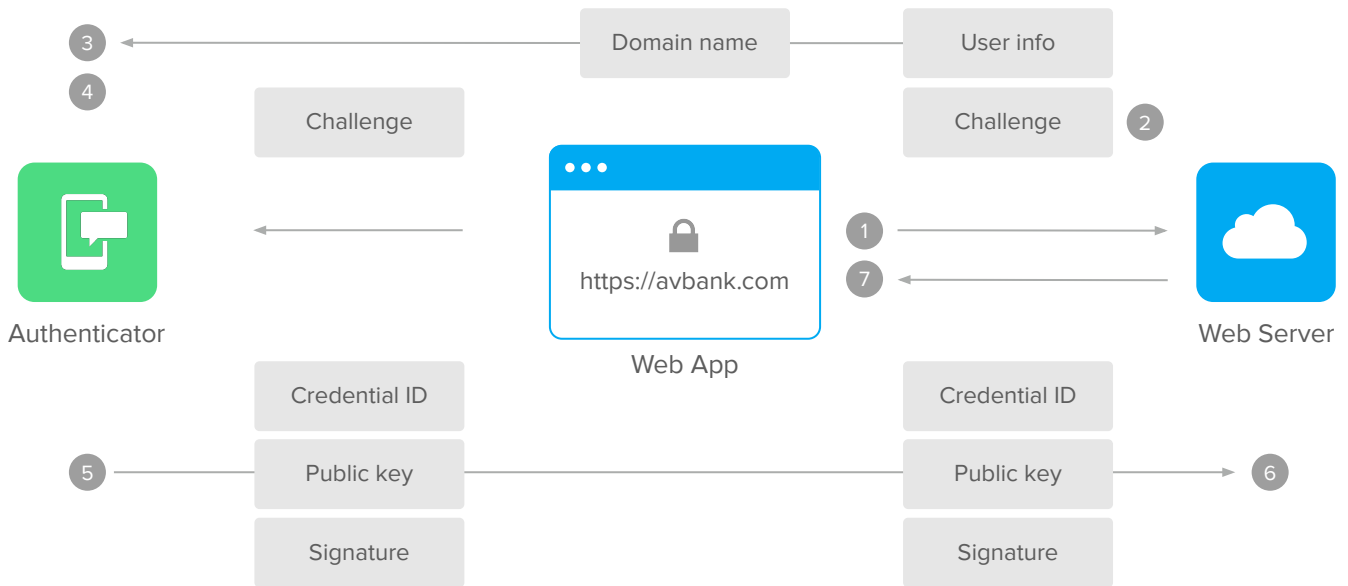
Main WebAuthn Flows



Enrollment Flow

Enrollment is when a user ties their authenticator (device) to the web server. Enrolling users' devices involves 3 main parties: the authenticator, the web app, and the web server. The steps are shown below:

Credential ID	q2we323d2rty123	User	Jason Sham
Private key	s12ds3d4s5da6	Challenge key	a9ds9dw9eds9d
Domain name	avbank.com	Credential ID	q2we323d2rty123
User info	Swaroop Sham	Public key	Ox3idfkek309



1. User initiates device setup on device
2. Web server generates a challenge key for registration (one time use)
3. Web server sends the following to the web app: Challenge Key, User Info
4. Web app adds authoritative domain name to information to be sent to authenticator
5. Authenticator asks for user consent
6. Once consent is given, authenticator stores Credential ID, Public/Private Key, User info, Domain name
7. Web app forwards the following to web server: Credential ID, Public Key, and Signature
8. Challenge Key is invalidated, and device is registered

Use Case 1—Enrollment Using Okta

With Okta, users are able to register a “platform” authenticator during the sign-in flow for your organization’s tenant. By configuring sign-on options and MFA enrollment policies, administrators can turn on web authentication for users via the feature flag. From the end user perspective, after entering their username and password, they are presented with an option to enroll for Windows Hello, Windows’ platform authenticator (Figure 1). This is Step 1 in the enrollment flow.

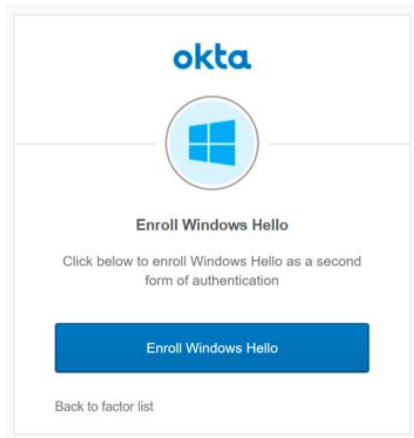


Figure 1

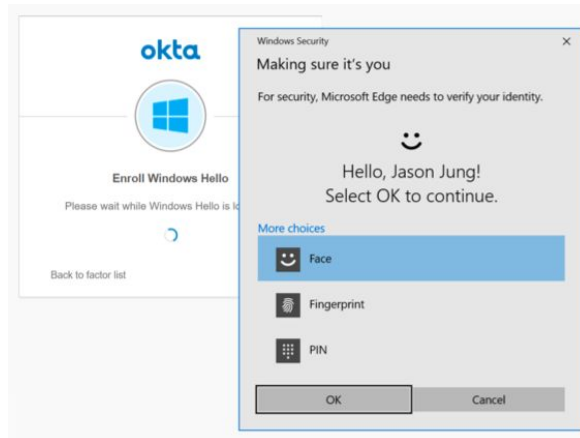
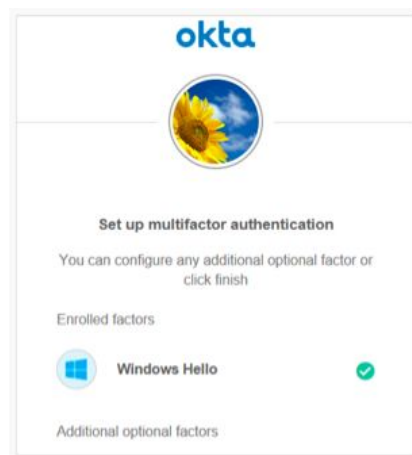


Figure 2

Assuming Windows Hello has been configured, there would be a one time set-up to create a public/private key (Step 5). Similar to the enrollment flow shown above, the user would give consent by using their face, fingerprint, or PIN that is stored on the local device.

Once consent is given in the form of a biometric factor or PIN, the credential ID, public key, and signature would then be sent back to the web server completing enrollment.



Okta's Best Practices

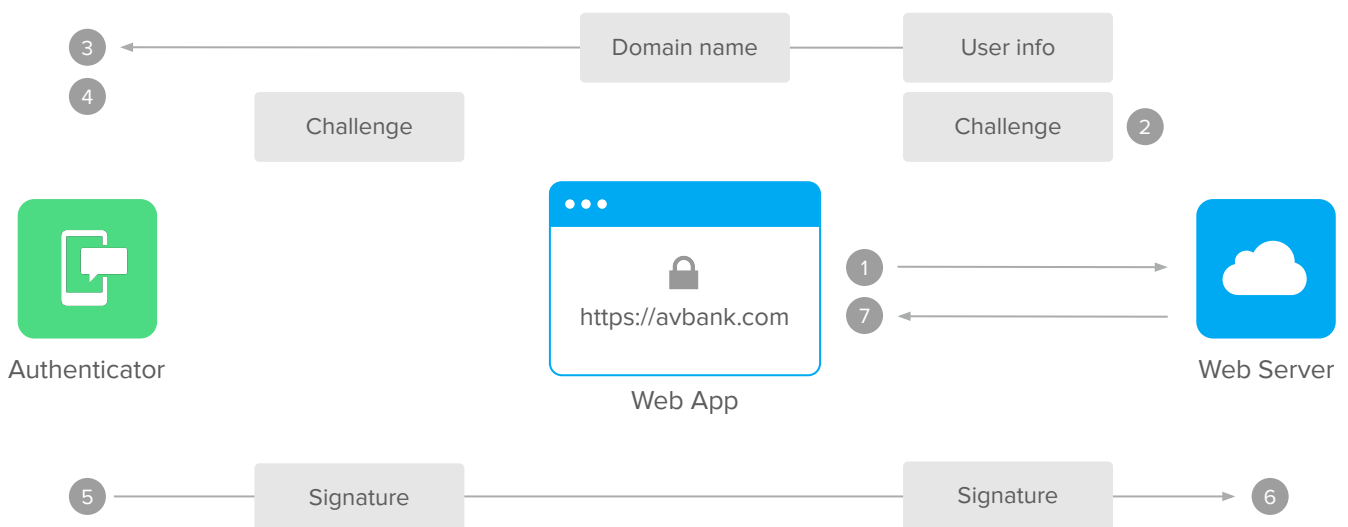
During enrollment, restricting initial enrollment to when users are within corporate networks, and limiting enrollment windows (e.g. new users must enroll within 3 days of their start date) can reduce risks during this phase.

In addition, features like end user visibility for factor enrollment and recovery, app specific enrollment policies (i.e. restricting enrollment when accessing certain apps), and supporting custom identity proofing processes during enrollment are other best practices you can use

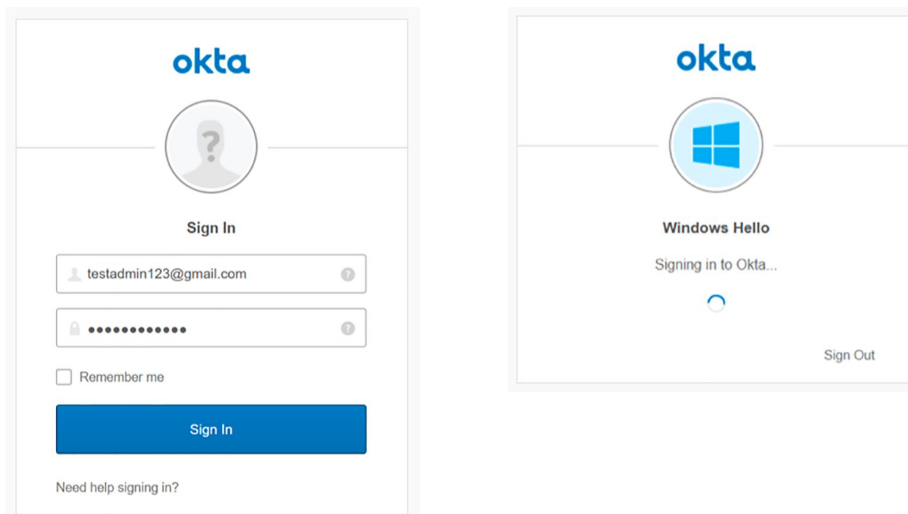
How Authentication Works

Now that the device has been enrolled, authenticating users with WebAuthn is seamless and secure for the user. Since enrollment has already been completed, this authentication flow mainly consists of the generation of a signature, based on the private key stored on the authenticator. This results in a challenge being generated each time.

Credential ID	q2we323d2rty123	User	Jason Sham
Private key	s12ds3d4s5da6	Challenge key	a9ds9dw9eds9d
Domain name	avbank.com	Credential ID	q2we323d2rty123
User info	Swaroop Sham	Public key	0x3idfkek309

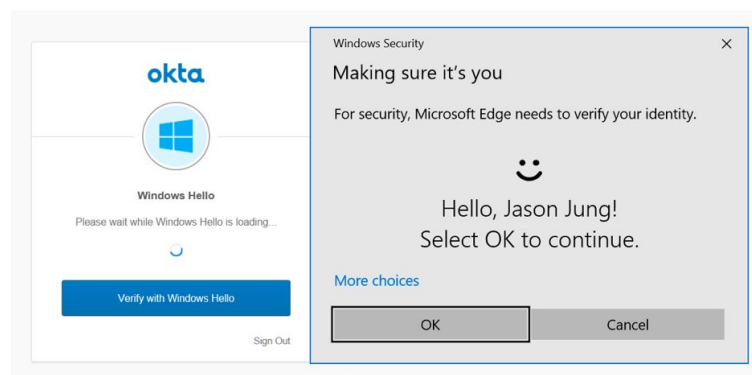


1. User initiates device setup on device
2. Web server creates a unique challenge that is sent to the authenticator
3. Authenticator receives challenge with domain name of challenge
4. Authenticator receives biometric consent from user
5. Authenticator generates cryptographic signature and is sent back to the web server
6. Web server verifies signature to unique challenge to login user
7. User is logged in



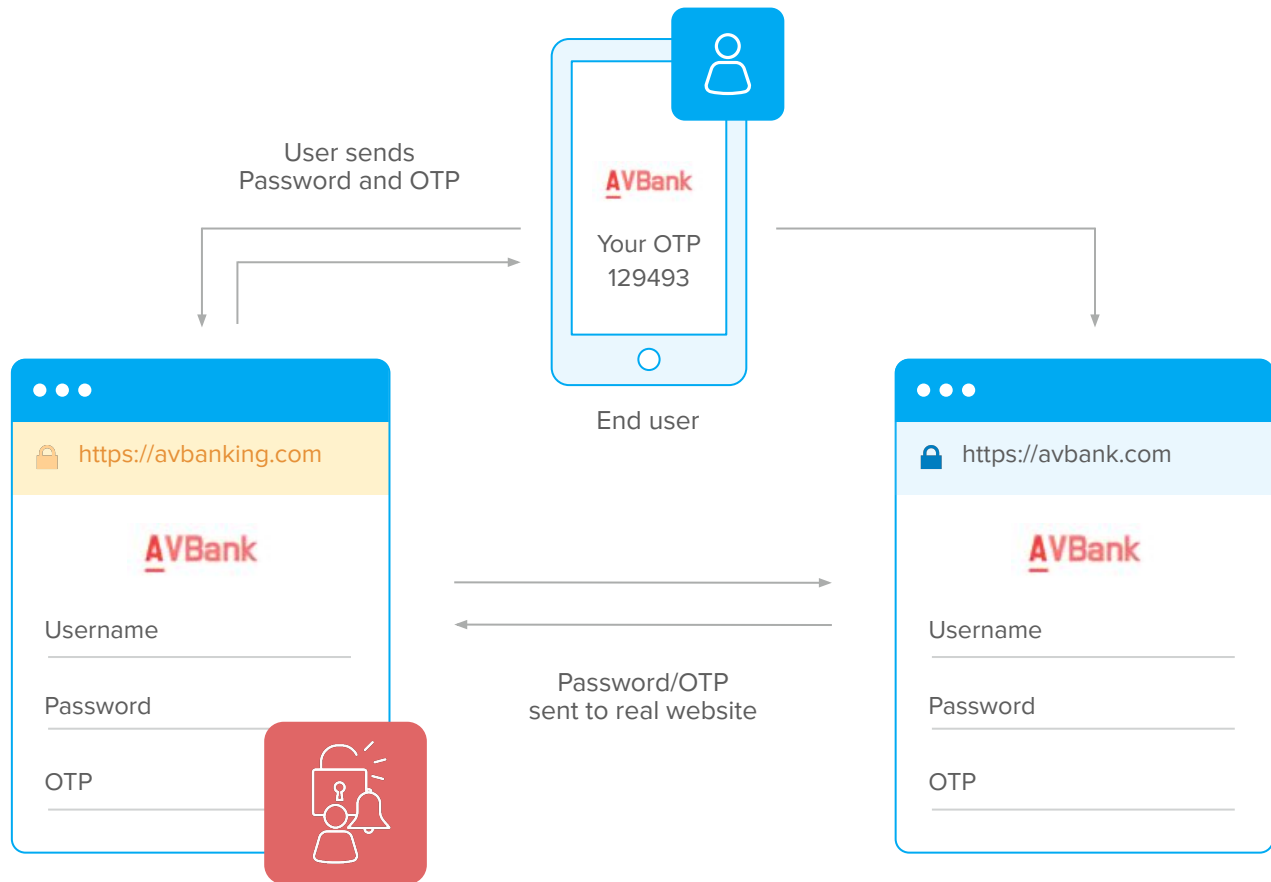
Use Case 2—Authentication with Okta

With Okta, users can use platform authenticators in place of step up authentication. After entering their username and password, WebAuthn can be used as an additional factor to ensure security. Okta can also be configured to challenge based on network zone, time of day, as well as type of device. To authenticate into Okta, the end user would be prompted for their Windows Hello verification.



WebAuthn and Phishing

The reason WebAuthn is resistant to phishing attacks is due to the domain name being stored on the authenticator. Since most phishing attacks are hosted on fake websites, the authenticator will compare domain names that were stored in Step 3.















Phishing Attack Example

When the challenge is received, the authenticator will check the domain name where the challenge originates. As shown above, typical phishing attacks usually redirect an end user to a fake website where they enter their credentials, which are then used for account takeover. When using WebAuthn, this risk is impossible due to the fact that the authenticator (or phone in this case) will verify the domain name for the user. This eliminates the risk of human error in entering credentials on a malicious website.

How Account Recovery Works

If you lose your device and are unable to use it for WebAuthn authentication into your instance, ensure you have other secure factors enrolled and set up for recovery. Some examples of other factors that are supported by Okta are:

Okta Verify 	SMS Auth 	Google Auth 	Windows Hello 
U2F (FIDO 1.0) 	YubiKey 	Duo Security 	Symantec VIP 
On-Prem MFA 	RSA SecurID 	Security Question 	Voice Call Auth 

Okta's Best Practices

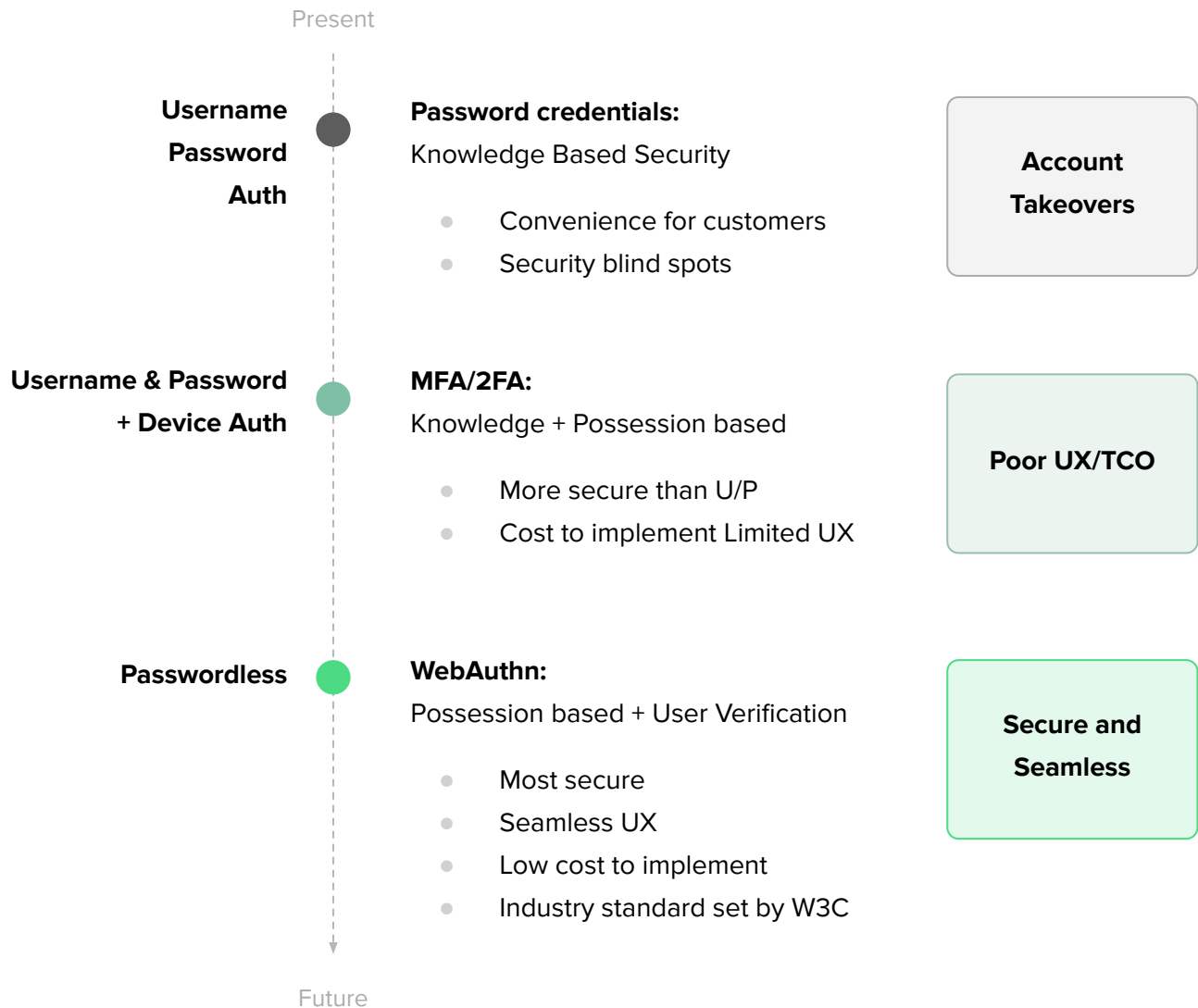
The security posture of your organization is only as good as your weakest security link. In the extenuating case where you have none of these factors set up, providing more visibility (e.g. through automated emails, notifying users when a factor recovery or password recovery is requested) ensures that suspicious incidents are immediately noticed and acted upon.

Part IV:

Moving Towards a Passwordless Future

With all these new WebAuthn flows, organizations can now move closer to a more secure, passwordless world. At Okta, we're dedicated to developing solutions to help our customers solve authentication challenges. We are also committed to supporting authentication standards like WebAuthn that help ease the way for broader adoption of passwordless strategies.

With this new standard, account takeovers and poor user experiences will be problems of the past as we improve the authentication experience through WebAuthn.



Okta offers a variety of products to help companies improve their security posture. Broken authentication practices have given rise to a range of identity attacks, and our [Adaptive Multi-Factor Authentication](#) solution is designed to mitigate these risks while minimizing the impact on the user.

For more on the technical specs of WebAuthn, check out [WebAuthn from W3C](#).

Other Helpful Okta Links

[“What is WebAuthn”](#)

[Developer’s guide: WebAuthn](#)

[Road to Passwordless](#)

[How FIDO2 + WebAuthn Offer a Seamless, Secure Login](#)

About Okta

Okta is the leading independent provider of identity for the enterprise. The Okta Identity Cloud enables organizations to securely connect the right people to the right technologies at the right time. With over 6,000 pre-built integrations to applications and infrastructure providers, Okta customers can easily and securely use the best technologies for their business. Over 6,100 organizations, including 20th Century Fox, JetBlue, Nordstrom, Slack, Teach for America and Twilio, trust Okta to help protect the identities of their workforces and customers.

www.okta.com