501CISO

501C CYBERSECURITY LEADERSHIP

Powered by ClearTone Consulting, LLC

Passkeys:

The New Security Standard

Prepared by

Brian Scott

PASSKEYS: THE NEW SECURITY STANDARD

Why Passkeys

Despite widespread adoption, Multi-Factor Authentication (MFA) is increasingly showing its limitations in the face of modern cyber threats. While MFA adds a critical layer beyond passwords, common implementations—such as SMS codes, authenticator apps, or push notifications—are still vulnerable to phishing, interception, SIM swapping, and user fatigue from constant prompts. Even more advanced forms like hardware tokens or biometric verification can be bypassed through social engineering or targeted attacks. These shortcomings highlight a growing problem: traditional MFA methods were not built to defend against today's industrialized and sophisticated attack landscape. To truly reduce risk while improving usability, the industry must move beyond MFA toward a more secure and seamless alternative—passkeys.

The MFA Dilemma

SMS-based MFA, one of the most widely used methods, is particularly vulnerable to several well-known attack vectors. Cybercriminals can intercept SMS codes through SIM swapping - a technique where an attacker convinces a mobile carrier to transfer a victim's number to a new SIM card they control. Once successful, the attacker can receive all SMS communications, including authentication codes. Additionally, SMS messages can be intercepted over insecure cellular networks or manipulated through malware on a user's device. Since phone numbers are often publicly associated with users or easily discoverable, attackers can readily target individuals with phishing or spoofed messages that appear legitimate but are designed to harvest MFA codes.

Authenticator apps, such as Google Authenticator or Microsoft Authenticator, improve security by generating time-based one-time passcodes (TOTPs) on the user's device. However, these apps are still susceptible to phishing attacks. Sophisticated phishing kits now mimic legitimate login pages, tricking users into entering their MFA codes in real time. Because TOTP codes are short-lived but not bound to specific sessions, an attacker can use them immediately to gain access. Moreover, users who back up their authenticator data to cloud storage or fail to secure their devices with strong PINs or biometrics risk exposure if their phone is lost or compromised.

Email-based MFA also introduces serious risks, particularly when email accounts themselves are protected by weak passwords or insufficient security controls. If an attacker gains access to a user's inbox - through phishing, credential stuffing, or poor password hygiene - they can easily retrieve one-time login codes or verification links. Furthermore, email is inherently slow and insecure, often traveling through multiple relays without encryption, making it a less-than-ideal medium for time-sensitive or secure communications. These vulnerabilities across all major MFA types demonstrate the need for a phishing-resistant, user-friendly alternative—one that cannot be easily intercepted, replayed, or reused. This is where passkeys offer a transformative leap forward.

Introducing Passkeys

Passkeys originated from a collaboration between major technology companies and standards organizations to solve the persistent security and usability challenges of passwords and traditional MFA. The foundation of passkeys lies in the **FIDO (Fast IDentity Online) Alliance** and the **World Wide Web Consortium (W3C)**, which developed the **WebAuthn** and **CTAP (Client to Authenticator Protocol)** standards. These protocols enable secure, phishing-resistant authentication using **public key cryptography**, where a device holds a private key and the online service stores a public key. The user's device verifies identity locally—via biometrics, PIN, or device unlock—without ever transmitting sensitive credentials.

The rollout of passkeys accelerated with strong backing from tech giants like Apple, Google, and Microsoft. In 2022, these companies jointly announced support for a common passkey standard that works across devices and platforms. Apple integrated passkeys into iOS and macOS, storing them in iCloud Keychain and syncing across devices. Google followed by enabling passkey support in Chrome and Android, while Microsoft added passkey functionality to Windows Hello and its authentication ecosystem. Today, passkeys are supported by major browsers, mobile platforms, and a growing number of websites and services—including PayPal, eBay, and Quickbooks—as the industry shifts toward passwordless, phishing-resistant authentication.

This momentum marks a significant turning point. As more services adopt passkeys and users become familiar with their simplicity and security, the traditional reliance on passwords and MFA codes will decline. Passkeys not only improve security by

eliminating shared secrets but also enhance the user experience with faster, frictionless logins that work across devices.

How Passkeys Work (WebAuthn Simplified)

Let's break down what's actually happening when passkeys are created and used. At the heart of it is a browser-based standard called **WebAuthn**, which allows a site to securely register and authenticate a user using a passkey instead of a password.

There are three main players in this process:

- The **Relying Party (RP)** – that's the website or service the user is trying to log into.
- The **Client** – usually the browser or operating system the user is using to access the site.
- The **Authenticator** – the secure hardware or software (like a phone or laptop) that holds the user's private key and performs cryptographic operations.

Registration

When a user registers on a site for the first time, this is what happens:

1. The user connects to the site (the Relying Party).
2. Their device creates a unique pair of cryptographic keys—one public, one private—specifically for that site.
3. The **private key** stays on the user's device, securely stored and tied to the site's domain.
4. The **public key** is sent to the site and saved alongside the user's account.

This setup means that even if the site's data is breached, the attacker only has a public key—which is useless without the private half stored securely on the user's device.

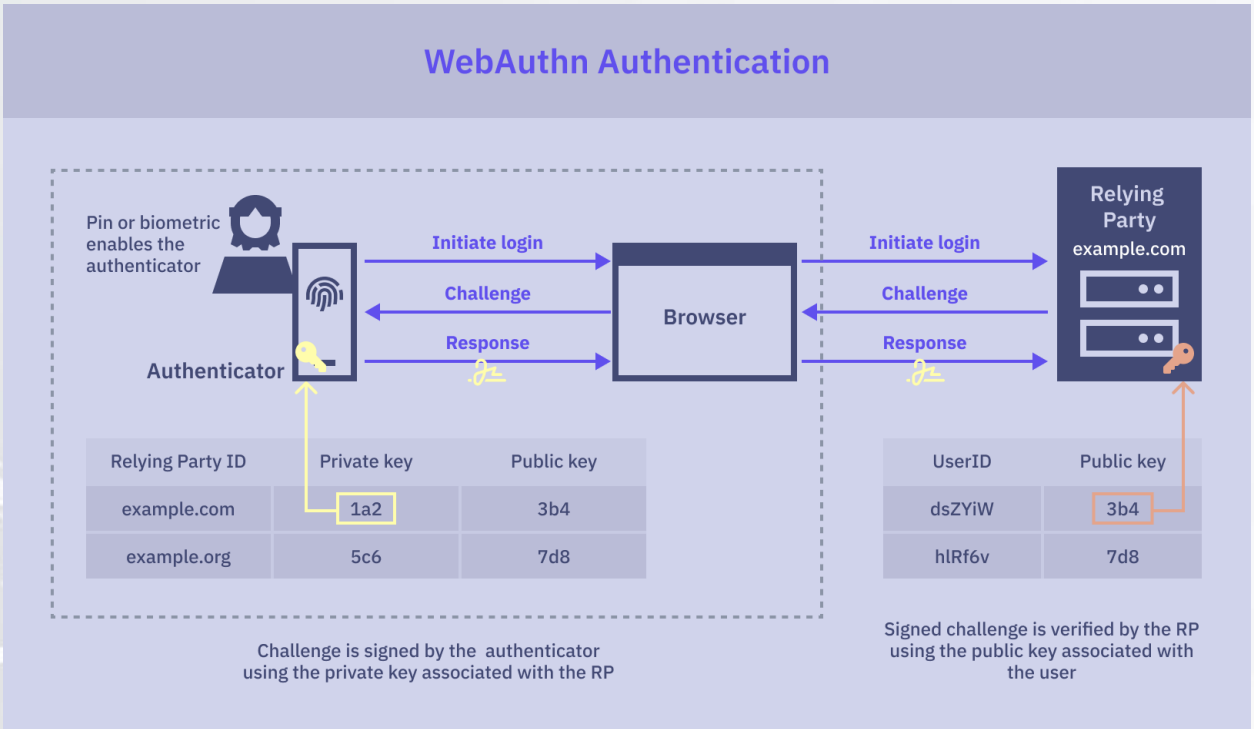
Authentication

The next time the user logs in, the process looks like this:

1. The user initiates a login.
2. The site (RP) generates a **one-time cryptographic challenge** and sends it to the user's device.

3. The user proves their identity—via a fingerprint, face scan, or PIN—to unlock their private key.
4. The private key signs the challenge.
5. That signed response is sent back to the site.
6. The site verifies it using the user's stored public key. If it checks out, access is granted.

What's powerful about this process is that it's phishing-resistant. The private key is locked to the real domain, so even if a user tries to sign into a fake site, the device won't allow the private key to be used. It's a secure, elegant replacement for passwords—and a much-needed step forward.



Integration with Password Management Systems

Password management systems are playing a crucial role in the widespread adoption of passkeys by making them easier to use across devices and platforms. Tools like **Apple iCloud Keychain**, **Google Password Manager**, and **1Password** have integrated passkey support, allowing users to generate, store, and sync their passkeys just like

they do with passwords. Because passkeys are stored securely within these systems and tied to a user's identity through biometrics or device unlock mechanisms, logging in becomes nearly instantaneous and requires no memory or manual entry.

Even more importantly, these password managers enable **cross-device and cross-platform convenience**. For example, a passkey created on a mobile device can be accessed on a laptop or tablet through cloud syncing, making it frictionless for users who frequently switch between devices. With support for Bluetooth proximity or QR-code-based authentication flows, even devices that don't share the same operating system can still participate in the passkey ecosystem. This compatibility and ease of use are helping reduce barriers to adoption, allowing users to enjoy stronger security without sacrificing convenience.

Wrapping It Up

Traditional MFA methods, while better than passwords alone, are increasingly outmatched by modern phishing techniques, SIM-swapping, and session hijacking. Passkeys offer a fundamentally stronger approach by removing shared secrets from the equation entirely. Built on open standards like WebAuthn and supported by major platform providers, passkeys provide a seamless login experience that is both more secure and more convenient than anything that came before. And with growing support from password managers and cloud syncing tools, deploying passkeys across your organization is more feasible than ever.

If you have questions about how passkeys work, how to implement them securely, or broader cybersecurity concerns for your organization, feel free to reach out to me directly. I'm always happy to help.