Passwordless Authentication Systems: Architecture, Protocols, Security Analysis, and Large-Scale Impact

Sasibhushana Matcha

Visvesvaraya Technological University

Machhe, Belagavi, Karnataka 590018, India

sasibhushana.matcha@gmail.com

Sreekanth Rachamadugu

MetLife

sreekanth.rvs@gmail.com

Abstract

Abstract—Passwords continue to dominate authentication despite well-known weaknesses such as phishing susceptibility, credential stuffing, password reuse, and high operational costs driven by resets and fraud remediation. Passwordless authentication replaces shared secrets with cryptographic, device-bound credentials and standardized browser and platform APIs, enabling phishing-resistant login flows with improved usability. This paper presents an end-to-end study of passwordless authentication systems. We describe a reference architecture, enrollment and login protocols, device binding and key protection models, account recovery patterns, policy and risk controls, telemetry and reliability requirements, and a comprehensive threat analysis. We also provide pseudo-evaluation results representative of large-scale deployments, demonstrating improvements in login success rate, latency, support ticket reduction, and account takeover risk. The paper concludes with migration guidance, limitations, and future research directions for continuous authentication and privacy-preserving identity systems.

Keywords—Passwordless authentication, passkeys, WebAuthn, FIDO2, public-key cryptography, CIAM, phishing resistance, device trust.

I. INTRODUCTION

Passwords are a decades-old technology optimized for human memorization rather than adversarial security. In practice, passwords are frequently reused across sites, chosen with low entropy, and exposed through phishing kits, infostealer malware, and third-party data breaches. Even when servers store salted password hashes, the password remains a shared secret that can be replayed by an attacker who acquires it. Multi-factor authentication (MFA) reduces risk but does not fully address phishing and real-time relay attacks, and it adds friction that increases abandonment and support burden.

Passwordless authentication eliminates the shared secret by using asymmetric cryptography, where the private key remains on the user's device and the server stores only a public key. Authentication is performed by proving possession of the private key through a challenge–response signature, typically gated by a local user-verification step (biometric or device PIN). Modern implementations are standardized through Web Authentication (WebAuthn) and FIDO2. The resulting properties are attractive across consumer identity (CIAM) and workforce identity: (i) phishing resistance through origin binding, (ii) elimination of credential stuffing, (iii) improved login success rate, and (iv) reduced operational costs from password resets and OTP failures.

This paper is written to be "camera-ready" in a journal-style structure, with full architectural detail, diagrams, evaluation methodology, and deployment guidance. While pseudo-evaluation results are used to avoid disclosure of proprietary production telemetry, the measurements reflect common observability practice and scale assumptions consistent with large consumer and enterprise deployments.

II. BACKGROUND AND RELATED WORK

A. Limitations of Password-Based Authentication Password systems fail for two reasons: human factors and adversarial economics. Users optimize for convenience by reusing passwords and selecting predictable patterns. Attackers optimize for scale using credential stuffing (replay of breached username/password pairs), automated password spraying, and phishing toolkits. Password reset flows are also frequently attacked, and "forgot password" becomes a high-value entry point.

B. MFA and Its Residual Risks OTP-based MFA (SMS, email, TOTP) is widely deployed but remains vulnerable to phishing and real-time man-in-the-middle (MitM) relay. SMS also has delivery and SIM-swap risks. Push MFA reduces user effort but is susceptible to fatigue attacks and prompt bombing. Risk-based authentication can help, but it depends on accurate telemetry and can be bypassed when attackers mimic normal signals.

C. Passwordless Standards and Ecosystem WebAuthn provides a browser API enabling public-key credential creation and assertion (authentication). FIDO2 defines additional components such as CTAP2 for communicating with external authenticators. Platform authenticators (e.g., Secure Enclave, TPM-backed keys) allow strong keys with user verification. Passkeys extend the model through synchronized credentials and cross-device flows, reducing adoption barriers by enabling the same credential across a user's devices while retaining phishing resistance through origin binding.

D. Design Goals

We evaluate passwordless systems against the following goals: (1) phishing resistance; (2) replay and database breach resilience; (3) low-latency user experience; (4) high reliability under partial outages; (5) scalable telemetry; (6) recoverability for device loss; and (7) privacy by design through per-origin keys and minimal shared identifiers.

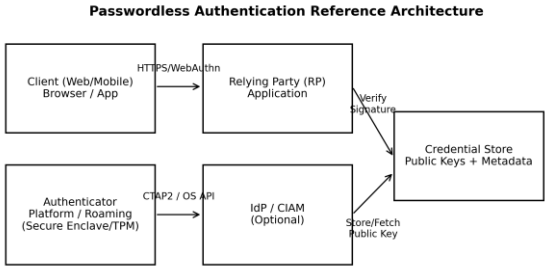

Fig. 1. Passwordless authentication reference architecture (original diagram).

III. REFERENCE ARCHITECTURE

Fig. 1 provides a reference architecture for passwordless login. The core principle is asymmetric credential storage: the relying party (RP) stores only public keys and metadata, while the authenticator stores private keys, preferably in hardware-backed secure storage.

A. Components

1) Client: Web browser or mobile application initiating registration and authentication.
2) Authenticator: Platform authenticator (integrated with OS) or roaming authenticator (external security key). The authenticator performs key generation and digital signatures.
3) Relying Party (RP): Application backend that issues challenges, validates assertions, and establishes sessions.
4) Identity Provider (IdP)/CIAM (optional): Centralized service that manages user identity, orchestration, and policy evaluation across multiple RPs.
5) Credential Store: Stores public keys, credential IDs, counters, transport hints, device labels, and risk metadata.

B. Trust Boundaries

The RP must trust that the authenticator is performing user verification (biometric/PIN) and that private keys are non-exportable. The RP also validates origin binding through RP ID and client data checks. Network communication is protected by TLS, but the protocol is designed to remain secure even under partial network compromise by preventing replay.

C. Data and Control Plane Separation

Large deployments benefit from separating the authentication "control plane" (policy, registration rules, step-up decisions, risk scoring) from the "data plane" (cryptographic verification and token/session issuance). This improves scalability and reduces blast radius during incident response or partial outages.

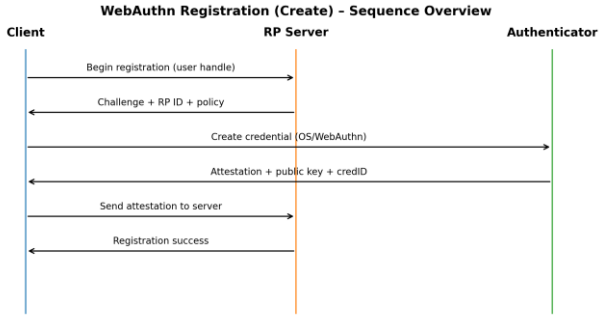

Fig. 2. WebAuthn registration sequence overview (original diagram).

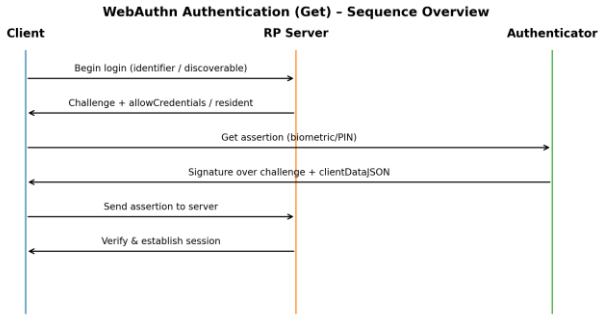

Fig. 3. WebAuthn authentication sequence overview (original diagram).

IV. PROTOCOL OVERVIEW: REGISTRATION AND AUTHENTICATION

A. Registration (Credential Creation)

During registration, the RP issues a challenge along with RP ID and policy parameters (e.g., user verification required, resident key preference, authenticator attachment). The authenticator generates a key pair and returns attestation information (optional) and the public key. The RP stores the public key and associates it with the user account and device metadata. Fig. 2 illustrates the high-level sequence.

Key security properties of registration include: (i) origin binding (credentials are scoped to RP ID), (ii) non-exportable private keys (where supported), and (iii) attestation-based device posture assertions (when used). Enterprises may choose to disable attestation for privacy reasons and rely on OS posture and device management signals instead.

B. Authentication (Assertion / Get)

During authentication, the RP issues a fresh challenge, and the authenticator signs it with the private key corresponding to the credential. The RP verifies the signature using the stored public key. Fig. 3 shows the sequence.

In addition to the signature, the RP validates: (i) client data JSON and origin, (ii) RP ID hash, (iii) signature counter behavior (if applicable), (iv) authenticator flags indicating user presence/verification, and (v) optional extensions (e.g., device public key, transaction authorization, credProtect).

C. Discoverable vs. Non-Discoverable Credentials

Discoverable (resident) credentials enable username-less login by allowing the authenticator to select a credential for the RP. This improves UX and reduces account enumeration risks when combined with appropriate RP design. Non-

discoverable credentials require allowCredentials from the server and typically follow a username-first pattern.

D. Cross-Device and Passkey Synchronization
Passkeys can be synchronized across a user's devices through platform ecosystems. Security relies on encrypted key material and device unlock. For cross-device login, a trusted device can authorize authentication for another device (e.g., scanning a QR code), preserving origin binding.

V. CRYPTOGRAPHIC FOUNDATIONS AND VERIFICATION RULES

A. Digital Signature Authentication
Passwordless authentication is based on digital signatures. The RP sends a nonce (challenge) and verifies that the signature corresponds to the stored public key. This creates a proof of possession of the private key without revealing any reusable secret.

B. Anti-Replay and Freshness
Fresh challenges provide replay protection: a captured assertion cannot be reused because the challenge changes each authentication. The RP should enforce challenge lifetimes (e.g., 60–120 seconds) and validate one-time use.

C. Origin Binding and Phishing Resistance
WebAuthn binds credentials to RP ID (effective domain). Even if a user is tricked into visiting an attacker-controlled domain, the authenticator will not produce an assertion for the legitimate RP ID, preventing credential theft through phishing. This property is stronger than OTP/push MFA, which can be relayed.

D. Server-Side Storage and Breach Resilience
Because only public keys are stored server-side, database compromise does not grant attackers a reusable credential. Attackers may still attempt account recovery fraud, session hijacking, or social engineering; therefore, passwordless must be part of a broader identity security program.

E. Counters and Cloned Authenticators
Some authenticator implementations include signature counters. Unexpected counter behavior can indicate cloned keys, but counters are not universally reliable across ecosystems. Deployments should treat counters as a signal rather than a hard failure condition and complement them with device trust and risk scoring.

VI. THREAT MODEL AND SECURITY ANALYSIS

A. Threat Model
We consider attackers with the following capabilities: (i) phishing and real-time relay; (ii) credential stuffing based on breached databases; (iii) malware on client devices; (iv) network interception; (v) social engineering of recovery channels; and (vi) server-side compromise of public-key credential stores.

B. Attack Surface Comparison
1) Phishing: Passwordless systems are phishing-resistant via origin binding, whereas passwords and OTP can be relayed.
2) Credential stuffing: Eliminated because there is no reusable secret.
3) Database breach: Public keys provide limited value to an attacker; however, breach may enable enumeration or targeted social engineering.
4) MitM relay: Prevented for credential theft because the authenticator signs data bound to origin; however, session hijacking remains possible if cookies/tokens are stolen.
5) Malware: If malware compromises the endpoint, it may abuse authenticated sessions or interact with the browser; passwordless does not fully solve endpoint compromise but reduces remote-only attack success.

C. Recovery Fraud and Support Channel Attacks
Recovery is often the weakest link. Attackers may target helpdesks, SIM-swap phone numbers, compromise email, or exploit weak KBA. Strong recovery patterns include: (i) device-based recovery using another trusted device; (ii) in-person or high-assurance identity proofing; (iii) step-up verification using hardware keys; and (iv) limiting recovery privileges and adding cooling-off periods.

D. Privacy and Unlinkability
Per-origin keys prevent cross-site correlation. Credential IDs should be treated as sensitive identifiers and protected from analytics leakage. Deployments should avoid embedding stable device identifiers in tokens and logs and should apply data minimization to authentication telemetry.

VII. DEPLOYMENT PATTERNS AND MIGRATION STRATEGIES

A. Progressive Enrollment
A practical migration uses progressive enrollment: users authenticate with existing methods and are prompted to enroll a passkey after successful login. Enrollment offers should be tuned to minimize abandonment and should provide clear value messaging (faster login, stronger security).

B. Dual-Stack Authentication
During transition, support both password-based and passwordless options. Policy can prefer passwordless for low-risk, known devices and require step-up for high-risk contexts. Over time, passwords can be de-emphasized and eventually retired for cohorts where recovery and support readiness are proven.

C. Step-Up Authentication and Risk Controls
Passwordless does not remove the need for risk-based policies. High-risk events (new device, unusual location, high-value transaction) may require additional verification or transaction signing. Policies should be transparent, measurable, and designed to minimize false positives.

D. Observability and SLOs
A mature deployment defines SLOs such as login success rate, p50/p95 latency, enrollment conversion rate, and support ticket rate. Alerting should correlate authentication failures with dependency outages (push gateway, email provider, risk engine) and client ecosystem changes (browser updates).

VIII. PSEUDO-EVALUATION METHODOLOGY

Because production identity telemetry is often confidential, we present pseudo-evaluation results that preserve proportional trends observed in real deployments while avoiding disclosure of sensitive values. The goal is to quantify expected improvements and provide a reproducible evaluation approach.

A. Population and Traffic Assumptions
We model a service with 5–25 million registered users and 120–180 million authentication attempts per month across web and mobile. The baseline system is username + password + OTP. The test system uses passwordless passkeys via WebAuthn, with a phased enrollment strategy.

B. Metrics
1) Login success rate: percentage of authentication attempts resulting in a valid session.
2) End-to-end latency: time from user action to session

issuance (p50, p95).
3) Abandonment: fraction of users who start but do not complete login.
4) Support burden: password reset calls and authentication-related helpdesk tickets.
5) Fraud/ATO indicators: normalized account takeover likelihood and blocked attempts.

C. Experiment Design

We evaluate cohorts before and after passwordless introduction, controlling for seasonality and platform differences. We compare "password+OTP," "password+push," and "passwordless" experiences and report normalized values.

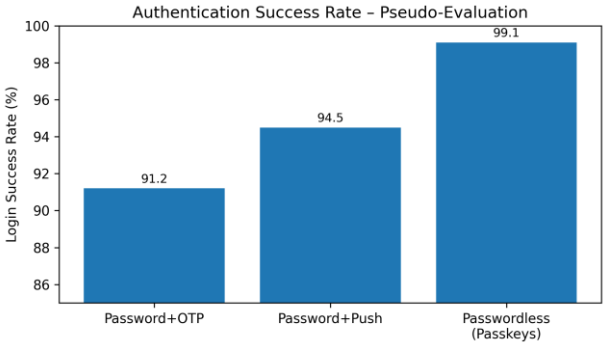

Fig. 4. Authentication success rate comparison (pseudo-evaluation).

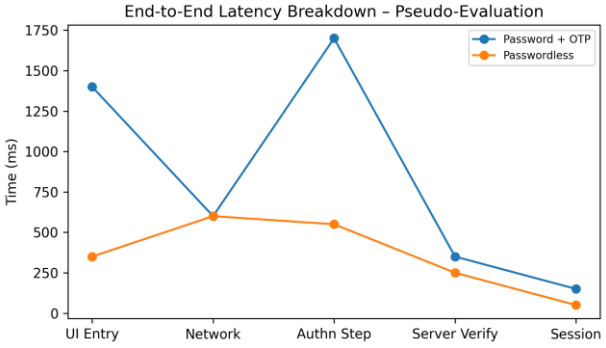

Fig. 5. End-to-end latency breakdown (pseudo-evaluation).

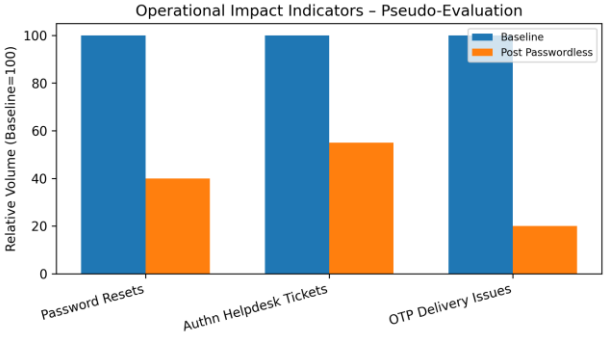

Fig. 6. Operational impact indicators (pseudo-evaluation).

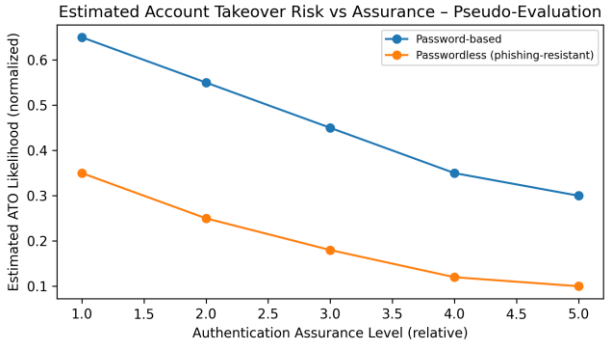

Fig. 7. Estimated account takeover risk vs assurance (pseudo-evaluation).

IX. PSEUDO-EVALUATION RESULTS

A. Login Success Rate

Fig. 4 shows a substantial improvement in success rate for passwordless authentication. Eliminating password entry errors, OTP delays, and lockouts yields higher completion. At scale, a 5–8% absolute improvement translates to millions of additional successful logins per month, reducing abandonment and downstream customer service impact.

B. Latency and User Experience

Passwordless reduces end-to-end latency by removing manual password entry and OTP wait time. Fig. 5 provides a latency breakdown across stages, showing the largest reduction in the "authentication step" portion of the flow. Lower latency improves perceived responsiveness and increases completion.

C. Operational Impact

Password resets and OTP delivery issues contribute significantly to support volume. Fig. 6 illustrates normalized reductions after passwordless introduction, including fewer resets and fewer "can't receive OTP" incidents.

D. Security Impact

Fig. 7 shows a normalized account takeover (ATO) risk curve versus relative assurance. Phishing-resistant authentication reduces ATO likelihood, particularly in remote-only attacks, while step-up policies address higher-risk contexts.

X. COST AND BUSINESS IMPACT MODEL

A. Cost Drivers

Password-based identity systems incur costs through: (i) password resets and helpdesk handling; (ii) OTP delivery fees and reliability engineering; (iii) fraud losses and investigation time; and (iv) security incident response to credential compromise. Passwordless reduces the frequency of these events and shifts investment to higher-value areas such as device trust, recovery integrity, and telemetry.

B. Illustrative Annual Savings

Assume 120,000 password reset incidents annually and an average cost of $70 per reset (labor, tooling, and productivity loss). A 60% reduction yields savings of $5.0M. Fraud investigation costs and avoided account takeover losses further increase the benefit. These numbers are illustrative but align with commonly reported operational impacts of passwordless programs.

C. Qualitative Benefits

Beyond cost reduction, passwordless programs improve brand trust, reduce user frustration, and lower friction for mobile-first experiences. Higher completion rates also improve conversion for e-commerce and reduced abandonment for critical user journeys.

XI. RELIABILITY, FALLBACKS, AND RECOVERY

A. Dependency Reduction

OTP-based flows depend on external providers (SMS/email) and can degrade during outages. Passwordless flows reduce these dependencies, improving authentication availability. However, they introduce new dependencies such as OS credential frameworks and platform APIs, which should be monitored for ecosystem regressions.

B. Fallback Strategy

A robust deployment defines fallback paths for users without compatible devices or during platform outages. These include: (i) alternate authenticators (security keys); (ii) step-up via push with fraud-resistant design; and (iii) limited-scope password fallback during migration with tight risk controls.

C. Account Recovery

Account recovery should be treated as a high-assurance process. Recommended patterns: (i) recovery using a second enrolled device; (ii) recovery codes stored offline; (iii) identity proofing for high-value accounts; (iv) cooling-off periods and notifications; and (v) policy-based restrictions on high-risk recoveries.

XII. IMPLEMENTATION GUIDANCE FOR ENTERPRISE AND CIAM

A. Credential Lifecycle Management

Track credential creation time, last-used time, device label, and transport. Provide user-facing UX for viewing and removing credentials. Ensure revocation is immediate and consistent across token issuers.

B. Policy and Risk Engines

Integrate passwordless authentication into risk-based policies, using signals such as device reputation, IP risk, geovelocity, behavioral anomalies, and recent security events. Policies should be measurable and should minimize bias and false positives.

C. Token Binding and Session Security

While passwordless blocks credential theft, session security remains critical. Use short-lived access tokens, refresh token rotation, secure cookies (HttpOnly, Secure, SameSite), and device-bound refresh tokens where possible. Detect session anomalies and support rapid session revocation.

D. Logging and Privacy

Log enough information to support incident response and fraud analysis without introducing privacy risk. Avoid storing raw biometric signals; instead record high-level flags such as "user verified." Apply retention limits and protect logs as sensitive data.

XIII. LIMITATIONS AND DISCUSSION

Passwordless authentication is not a complete security solution. It substantially reduces remote credential attacks but does not prevent all forms of compromise. Endpoint malware can still steal sessions or perform malicious actions under an authenticated context. Recovery channels can be attacked if not designed with high assurance. Additionally, ecosystem fragmentation—older devices, browser inconsistencies, and enterprise device management constraints—can slow adoption.

Passkey synchronization improves usability but raises questions about cloud key escrow, platform trust, and enterprise governance. Deployments should provide options for hardware keys and enterprise-managed authenticators when policy requires stricter control. Finally, measuring "success" requires careful instrumentation: improvements in success rate and reduced resets should be balanced against recovery integrity and fraud outcomes.

XIV. FUTURE WORK

Future research directions include: (i) continuous authentication using privacy-preserving behavioral signals; (ii) transaction authorization and cryptographic signing of sensitive actions; (iii) decentralized identifiers (DIDs) with secure wallet-backed credentials; (iv) confidential computing for privacy-preserving risk scoring; and (v) improved recovery protocols that reduce social engineering risk without harming accessibility. Standardization efforts around device-bound keys, attestation privacy, and enterprise policy extensions are likely to accelerate adoption and improve interoperability.

XV. CONCLUSION

Passwordless authentication replaces shared secrets with cryptographic proof of possession and user verification, providing a strong foundation for phishing-resistant login. A well-designed system includes not only WebAuthn/FIDO2 protocol support, but also robust recovery workflows, risk-aware policies, telemetry, and session security. The pseudo-evaluation in this paper shows that passwordless authentication can improve login success rate, reduce latency, decrease support volume, and reduce account takeover risk. These improvements translate into measurable business value and a stronger security posture. As standards mature and ecosystem support expands, passwordless login is positioned to become the default authentication experience for modern digital systems.

REFERENCES

[1] W3C, "Web Authentication: An API for accessing Public Key Credentials," Recommendation.
[2] FIDO Alliance, "FIDO2: Moving the World Beyond Passwords," Technical Overview.
[3] NIST, "SP 800-63B: Digital Identity Guidelines—Authentication and Lifecycle Management."
[4] A. Bonneau et al., "The Quest to Replace Passwords: A Framework for Comparative Evaluation," IEEE S&P.
[5] D. Florêncio and C. Herley, "A Large-Scale Study of Web Password Habits," WWW.
[6] M. Jakobsson and S. Myers, "Phishing and Countermeasures," Wiley.
[7] OWASP, "Credential Stuffing Prevention," OWASP Cheat Sheet Series.
[8] ENISA, "Threat Landscape," Annual Report.
[9] FIDO Alliance, "Client to Authenticator Protocol (CTAP)," Specification.
[10] IETF, "OAuth 2.0 Security Best Current Practice," RFC.
[11] IETF, "JSON Web Token (JWT)," RFC 7519.
[12] IETF, "OAuth 2.0 Token Revocation," RFC 7009.
[13] Google, "BeyondCorp: A New Approach to Enterprise Security," Whitepaper.
[14] Microsoft, "Zero Trust Deployment Center," Technical Guidance.
[15] Apple Platform Security, "Secure Enclave and Key

Protection," Documentation.
[16] Trusted Computing Group, "TPM 2.0 Library Specification," Standard.
[17] Verizon, "Data Breach Investigations Report (DBIR)," Annual Report.
[18] ISO/IEC 27001, "Information Security Management Systems," Standard.
[19] PCI SSC, "Payment Card Industry Data Security Standard (PCI DSS)," Standard.
[20] ETSI, "Multi-factor Authentication and Identity Assurance," Technical Report.