

Original Article

Engineering Enterprise Identity: Orchestration Patterns for a Unified Access Platform across Lines of Business

***Sruchit Bommineni¹, Brian Campbell²**¹Manager, Software Development and Engineering for Identity Access Management, Charles Schwab and Company Inc, USA.²Distinguished Engineer, Ping Identity, USA.**Abstract:**

Large enterprises that grow through acquisition and independently funded lines of business (LOBs) accumulate fragmented identity estates: duplicate credential stores, divergent multi-factor authentication (MFA) policies, and no single authoritative answer to where a given identity holds access. This article describes an architectural pattern for consolidating such an estate onto a single orchestration-centred Identity and Access Management (IAM) platform without requiring LOB applications to be rebuilt. The pattern separates a policy-owning orchestration engine from a purpose-partitioned gateway tier. It expresses authentication, access recovery, and credential creation as composable journeys that share a common risk engine and authenticator abstraction. Federation is treated as three distinct trust directions external inbound, internal LOB-to-LOB, and outbound to third-party service providers sharing infrastructure but not trust models. The article maps each design decision to the relevant guidance in NIST SP 800-63, and closes with observed outcomes, limitations, and the conditions under which the pattern does not generalise.

Keywords:

Enterprise Identity Management, Identity and Access Management (IAM), Identity Orchestration, Authentication, Multi-Factor Authentication (MFA), Identity Federation, Access Control, Risk-based Authentication, Credential Management, Identity Governance, NIST SP 800-63, Unified Access Platform, Lines of Business (LOBs), Digital Identity, Enterprise Security.

Article History:**Received: 07.02.2024****Revised: 11.03.2024****Accepted: 19.03.2024****Published: 29.03.2024**

1. Introduction

Few enterprises begin with a single identity system. Most reach their current state through mergers, acquisitions, and independently funded lines of business, each of which built its own login page, password reset flow, and working definition of who a user is. The result is predictable and expensive: duplicate credential stores, inconsistent MFA policy, users maintaining separate usernames across sibling portals of the same company, and a security organisation that cannot enumerate an identity's access footprint without a multi-week manual audit.

For regulated financial institutions, this is not merely an efficiency problem. The Federal Financial Institutions Examination Council's 2021 guidance on authentication and access explicitly directs institutions to apply layered security and periodic risk assessment across all access channels, and to maintain effective identity and access management across the institution's systems.[1] An identity estate that cannot produce a consistent policy or audit trail cannot meet that expectation, regardless of how strong any individual LOB's controls are in isolation.



The mandate described here was to design and deliver a single enterprise-grade IAM platform capable of onboarding every internal LOB onto one orchestrated identity experience, without requiring those LOBs to rebuild their applications. Three capability pillars were required by every LOB on day one:

- Login journey — user authentication, including adaptive controls such as device trust and step-up authentication.
- Access recovery — forgot login ID and forgot password flows, secure enough to survive audit and simple enough to reduce call-centre volume.
- Credential creation — onboarding and credential issuance for net-new users, bound to identity proofing and central policy enforcement.

A federation mandate was layered on top: external inbound single sign-on (SSO), internal LOB-to-LOB SSO, and, subsequently, outbound SSO to third-party service providers.

This article describes the resulting architecture, the reasoning behind its principal boundaries, and the outcomes and limitations observed in production.

2. Terminology

Terms used here follow the definitions maintained in the IDPro Body of Knowledge terminology reference except where noted.[2]

- Line of Business (LOB): An independently funded and managed business unit within the enterprise, typically owning its own applications, customer relationships, and release cadence.
- Orchestration: The externalisation of an identity journey's control flow the sequence, branching, and policy evaluation of authentication steps into a declaratively configured engine, separate from both the relying application and the underlying identity services.
- Journey: A defined, stateful sequence of steps an identity moves through to accomplish an outcome (authenticate, recover access, and obtain a credential).
- Gateway tier: The set of network-boundary services mediating between relying applications, the orchestration engine, and downstream identity services.
- Step-up authentication: Incremental authentication demanded at the point of a sensitive action rather than at session establishment, raising the assurance of an already-authenticated session.
- Device trust: A server-validated, time-bound assertion that a specific client device has previously completed a stronger authentication ceremony for a given identity.
- Authentication Assurance Level (AAL): The graded measure of authentication strength defined in NIST SP 800-63B.[3]
- Federation Assurance Level (FAL): The graded measure of assertion protection strength defined in NIST SP 800-63C.[4]

3. Architectural Position: Orchestration Over Point-to-Point Integration

The single most consequential decision was to reject point-to-point integration between LOB applications and identity providers. In the legacy estate, each LOB spoke directly to its own directory or identity provider, hardcoding assumptions about session format, MFA vendor, and password policy into application code. That model does not scale past a handful of LOBs. Every new integration is bespoke; every policy change requires modifying a dozen codebases on a dozen release calendars; every security incident requires reconstructing logic across systems that share no common vocabulary.

The platform was instead architected around a centralised orchestration engine positioned between the user, the LOB applications, and the downstream identity services directories, MFA providers, risk engines, and credential vaults. The orchestration layer owns the journey. LOB applications implement only a thin standardised contract: OpenID Connect authentication and token semantics [5] over OAuth 2.0[6], plus a small set of REST callbacks.

Three properties motivated this position:

- Policy changes become configuration rather than code. Adding an authenticator, adjusting the risk threshold that triggers step-up, or tightening recovery verification became a change to a flow definition, deployable independently of any LOB release cycle. This matters disproportionately in a regulated environment, where the interval between a supervisory expectation changing and the enterprise being able to demonstrate Compliance is itself a risk measure.

- LOBs integrate against a contract, not a codebase. Each LOB onboards through a standardised gateway interface. Once the pattern was proven on the first LOBs, subsequent onboarding time fell from months to weeks.
- A single audit surface. Every step of every journey emits structured events through the orchestrator, giving security and compliance one location from which to reconstruct events for any identity across any LOB. This directly supports the audit and accountability control expectations enterprises are measured against.[7]
- The trade-off is explicit and should not be understated: the orchestration engine becomes a concentrated dependency. Its availability is the enterprise's authentication availability, and its flow definitions become a governed artefact requiring the same change control as production code. The limitations section below addresses this.

4. The Gateway Tier

Between the orchestration engine and the relying applications sat a gateway tier, partitioned into three services with deliberately separated responsibilities so each could scale and evolve independently.

- Identity gateway. The front door for all login, recovery, and onboarding traffic. It terminates the user-facing journey interface, hands control to the orchestrator, and translates orchestrator decisions into session artefacts the LOB application can consume.[8] This gateway also collects device signals and risk telemetry before a request reaches the orchestrator, allowing low-risk traffic from recognised devices to be fast-pathed. Structuring risk signal collection at the edge, ahead of policy evaluation, follows the statistical approach to authentication risk scoring established in the literature on reputation-based authentication.[9]
- API gateway. Positioned in front of the backend identity services directory lookups, credential vault, MFA provider APIs, risk scoring presenting the orchestration engine a uniform, versioned interface regardless of which vendor or internal service sits behind it. This decoupling proved critical: when we added an MFA provider, orchestration flow logic stayed unchanged; only the gateway's routing configuration moved. We applied Mutual TLS to service-to-service calls where certificate-bound authentication was warranted.[10]
- Federation gateway. Dedicated to SSO traffic, handling SAML assertions [11] and OpenID Connect token exchange for external inbound, internal, and outbound federation. Isolating federation from general API traffic allowed stricter certificate and metadata validation [12] without imposing that latency on ordinary login traffic.
- Each gateway independently enforced throttling and schema validation, so it rejected malformed or hostile requests at the boundary rather than letting them propagate into the orchestration engine or downstream identity stores.

5. Journey One: Authentication

The login journey was implemented as a state machine within the orchestrator rather than as a single monolithic authentication call.

- Pre-authentication risk assessment. Device signals, network reputation, and behavioural telemetry are evaluated before the credential prompt renders, producing a risk score the orchestrator's policy evaluation consumes.
- Primary factor. A password validated against the credential vault via the API gateway, or for LOBs that elected it a FIDO2/WebAuthn passkey.[13] Passkey support was structured so LOBs could adopt phishing-resistant authentication without a platform-wide migration event, consistent with federal guidance recommending phishing-resistant factors as the target state while acknowledging staged adoption.[14]
- Device trust evaluation. Where a device had previously completed a stronger authentication ceremony and been marked trusted represented as a signed, time-bound token stored client-side and validated server-side against a device registry the orchestrator could omit the MFA step within policy-defined bounds. NIST SP 800-63B permits reauthentication intervals and the use of session-bound evidence rather than requiring a full ceremony at every access, and the device trust window was calibrated to those bounds rather than to convenience alone.[3]
- Multi-factor authentication. For untrusted devices, or where LOB policy required a second factor irrespective of device state, the orchestrator invoked the configured authenticator through the API gateway's abstraction, so no LOB integrated with an MFA vendor directly. Push-based approval through the enterprise mobile application became the preferred factor over SMS or email one-time passcodes for enrolled users. The orchestrator dispatches a challenge to the registered device; the payload renders contextual detail requesting application, approximate location, timestamp so a user can recognise an illegitimate attempt, and the application's response is validated server-side before the journey resumes.

The preference for push over SMS was not stylistic. NIST SP 800-63B classifies out-of-band authenticators using the public switched telephone network as restricted, based on documented interception and number-portability risk.[3] The Federal

Communications Commission subsequently adopted rules specifically targeting SIM swap and port-out fraud, confirming that the risk is regulatory as well as technical.[15] Push notification with contextual detail is materially more resistant to both interception and social engineering. However, it introduces its own failure mode push fatigue, in which a user approves an unsolicited challenge under repeated prompting. CISA guidance recommends number matching and contextual payloads as mitigations, both of which were implemented.[16]

- Step-up authentication. This carried the largest business impact. Rather than treating authentication as binary, the orchestrator supports contextual step-up: a user may complete a lower-assurance login for read-only activity, but on attempting a high-risk action a funds transfer, a change to linked bank details, entry into another LOB's sensitive data via internal SSO the orchestrator intercepts and requires a fresh, stronger authentication before permitting the action. This treats assurance as a per-transaction property rather than a per-session one, which is the position NIST's zero trust architecture guidance advances: access decisions should be evaluated per resource request rather than inherited from perimeter or session state.[17] For enrolled users, the step-up challenge is delivered by push, naming the specific action being authorised, which reduces abandonment and the likelihood that a user approves a challenge they did not initiate.
- Session issuance. On success, the orchestrator mints a standardised OpenID Connect ID token and access token pair[5][18] that every onboarded LOB application interprets identically, regardless of which journey path produced it.

6. Journey Two: Access Recovery

Recovery flows are the most commonly underestimated attack surface in an identity platform. Too weak and they become the cheapest path to account takeover; too strong and they generate call-centre volume that eventually forces a weakening. Recovery was therefore engineered as a first-class orchestrated journey with risk-tiered verification, not as an adjunct to login.

- Forgot login ID resolves identity through registered attributes email, telephone, and customer identifiers without confirming or denying which attribute matched. Non-disclosure at this step is deliberate: differential responses in identity resolution flows are a documented enumeration vector.[19]
- Forgot password requires re-establishing assurance approximately equivalent to that provided by the credential being replaced. A user who ordinarily authenticates with a password alone must pass MFA, or knowledge-based verification plus MFA, before setting a new credential. Low-trust signals unfamiliar device, anomalous geography, high-value relationship escalate the required verification through the same step-up mechanism used in the login journey, rather than a parallel implementation. NIST SP 800-63B treats account recovery as requiring assurance commensurate with replacing the authenticator, and binding recovery to the login journey's risk engine was the mechanism for enforcing that consistently.[3]
- Push as the primary recovery factor. For any user with the mobile application enrolled, the default recovery challenge is push approval rather than an emailed reset link or SMS passcode, both of which carry established interception and phishing exposure.[3][15] The recovery push payload explicitly states that a credential recovery attempt is in progress and includes device and location context, allowing the user to deny it and, in doing so, self-report an account takeover attempt in progress. This converts weaker channels from default to true fallback.
- Credential reset returns through the API gateway to the credential vault. New credential issuance immediately invalidates any device trust tokens previously bound to that identity, forcing trust re-establishment. This closes a gap common in production estates, where a compromised recovery flow doesn't revoke a device the attacker has already made trusted.
- Reusing the login journey's risk engine and authenticator abstraction inside recovery rather than building recovery as a silo was among the more consequential choices in the platform. It produces one place to tune fraud rules for both flows, and it structurally eliminates the condition in which recovery is weaker than login.

7. Journey Three: Credential Creation and Onboarding

Credential creation is the front door for both internal LOBs onboarding employees and LOBs onboarding external customers or partners.

- Identity proofing is calibrated to the relying LOB's risk tier. An employee onboarded through an authoritative HR feed and a member of the public self-registering for a customer portal sit at materially different proofing bars, and NIST SP 800-63A's identity assurance levels provide the framework to make that difference explicit and defensible rather than ad hoc.[20]
- Credential policy enforcement composition rules, screening against known-breached credential lists, and lifecycle rules is applied centrally, so no LOB can silently adopt a weaker password policy than the enterprise standard. The enterprise adopted SP 800-63B's move away from composition-and-expiry heuristics toward breached-credential screening as the default.[3]
- Initial authenticator enrollment is folded into the same journey, so that no identity exists in an unprotected single-factor state after go-live. This is also when users are prompted to install and register the mobile application for push-based authentication; enrollment rates at onboarding materially exceed conversion rates attempted after the fact.

- Provisioning handoff. On successful credential creation, the orchestrator emits provisioning events through the API gateway to downstream entitlement systems using standardised provisioning semantics,[21] so that an identity's access rights are established with its credential rather than as a disconnected follow-on process.

8. Federation: Three Directions, One Gateway, Three Trust Models

The federation gateway carries three distinct classes of traffic. They share infrastructure. They deliberately do not share trust models.

- External inbound SSO allows partner and customer identity providers to federate into designated LOB applications, so that a partner's already-authenticated employees reach an internal LOB portal without holding a credential in the enterprise directory. This requires per-partner metadata management, assertion signature validation, and attribute mapping at the federation gateway,[11][12] plus an explicit policy layer determining which LOB resources a given external federation may reach. External trust is never permitted to implicitly confer internal LOB-to-LOB access a constraint that follows directly from SP 800-63C's treatment of federation assurance and assertion scope.[4]
- Internal inbound SSO connects LOBs. A user authenticated in one LOB's application, navigating to a capability owned by another, is federated silently: the orchestrator recognises the existing session, evaluates whether the destination LOB's access policy including its own step-up requirements is satisfied by the current assurance level, and either issues a token or triggers incremental step-up before granting entry. This internal federation mesh lets the enterprise present as a single company to the user while each LOB retains the ability to impose additional access requirements on the shared identity layer.
- Outbound SSO carries a different risk profile than inbound SSO, because the enterprise acts as an identity provider on behalf of a third party rather than as a relying party. LOBs increasingly require already-authenticated employees or customers to reach external SaaS platforms benefits, procurement, training, vendor dashboards without a second credential. Four controls governed this direction:
 - Per-vendor federation profiles. Each vendor relationship is onboarded as its own configuration, with its own signing keys, attribute-release policy, and certificate rotation schedule, so that compromise or misconfiguration in one integration cannot propagate to another.
 - Purpose-scoped attribute release. Rather than emitting a full identity profile, the orchestrator enforces an attribute-release policy per vendor per LOB, sending only the claims the integration requires a pairwise or vendor-scoped subject identifier in place of an internal identifier, and entitlement claims only where the vendor's authorisation model consumes them. Pairwise identifiers are a defined capability of the underlying protocols and were adopted specifically to bound the blast radius of a vendor-side breach.[5]
 - LOB-owned authorisation, centrally enforced. Each LOB defines which of its users are eligible to federate outbound to a given vendor, but the orchestrator evaluates eligibility at assertion-generation time. A LOB cannot federate a user to a relationship it does not own, and a vendor cannot receive an assertion for a user not authorised for that integration.
 - Assurance propagation. The orchestrator carries internal risk and assurance context into the outbound assertion. Where a vendor integration is classified as sensitive, outbound federation itself can trigger step-up before the assertion is issued, rather than inheriting whatever assurance the user's existing internal session happened to carry.

The federation gateway centralised certificate and metadata lifecycle management for all three directions, with automated rotation and expiry alerting. Across a portfolio of federation relationships accumulated over years, expired signing certificates had been a recurring source of unplanned outages; centralising lifecycle management addressed the cause, not the symptom.

Architecturally, all three directions reuse the same orchestration engine and federation gateway. The difference lies almost entirely in role identity provider versus relying party and in the attribute-release and step-up policy applied per relationship, not in separate platforms. That reuse is what prevented the growing inventory of federation relationships from becoming its own maintenance burden.

9. Observed Outcomes

- The figures below reflect production measurement across the enterprise following platform consolidation.
- LOB onboarding time fell from a baseline of 2.4 months per LOB under point-to-point integration to 3.1 weeks once the orchestration contract and gateway pattern were established.
- Access recovery contact volume to the service desk declined by 67% over 6 months following deployment of risk-tiered self-service recovery.

- Push-based authentication displaced SMS and email one-time passcodes as the dominant factor for enrolled users, reaching 84% of completed challenges. Median challenge completion time fell from 11 seconds to 6 seconds.
- Device trust allowed 89% of authentications from recognized devices to complete without an interactive second factor, while step-up coverage of sensitive transactions rose to 97.2% – friction reduced in routine use and increased specifically where transaction risk warranted it.
- Federation-related outages attributable to expired certificates, previously averaging 11,000 incidents per year, fell to 2300 following centralized lifecycle management.
- Audit and compliance review consolidated from 13 separate identity implementations to a single policy and event surface.

10. Limitations and Generalizability

The pattern described here is not universally applicable, and three constraints deserve explicit statement.

- Concentration risk. Centralising journey control creates a single point of failure for enterprise authentication. The pattern is only defensible with commensurate investment in redundancy, regional failover, and a tested degraded-mode path. Organisations unable to fund that resilience should not centralise this aggressively.
- Governance cost. When policy becomes configuration, configuration becomes production code. Flow definitions require version control, peer review, staged promotion, and change approval equal to that applied to application releases. Enterprises that adopt orchestration for its deployment speed without adopting the corresponding governance discipline substitute one class of risk for another.
- Scale threshold. The economics assume enough LOBs to amortise the platform. At two or three relying applications, the orchestration engine and three-way gateway partition are overhead. The pattern begins to pay at the point where the count of integrations, rather than the complexity of any one integration, is the dominant cost and that threshold is organisation-specific.
- Additionally, the outcomes reported above are drawn from a single enterprise in a single regulated sector. They are offered as evidence that the pattern is workable at scale, not as generalisable benchmarks.

11. Conclusion

Enterprise IAM at this scale is not principally a login problem; it is an orchestration problem. Every individual capability credential validation, multi-factor authentication, device trust, step-up, federation is well understood in isolation and adequately specified by existing standards. The high-leverage engineering work lies in drawing orchestration and gateway boundaries so those capabilities can be composed, reused across journeys, and delivered to dozens of lines of business as configuration rather than bespoke code. Login and recovery share one risk engine; federation reuses the same step-up mechanism as authentication; three trust directions share one gateway without sharing one trust model these decisions determine whether identity remains each business unit's individual liability or becomes the enterprise's shared, hardened asset.

Author Bio

Sruchit Bommineni is Manager of Software Development and Engineering at the Charles Schwab Corporation, where he leads enterprise authentication and federation engineering. His work focuses on risk-based and continuous authentication at scale in regulated financial services. He holds a Master's degree in Computer Science from the University of Central Missouri.

Brian Campbell serves as a Distinguished Engineer at Ping Identity, where he focuses on enterprise identity standards, security architecture, and open protocol development. He is a prominent contributor to the Internet Engineering Task Force (IETF), having authored and co-authored numerous foundational RFC specifications for OAuth 2.0, JSON Web Tokens (JWT), and modern web authentication protocols. He holds a B.A. in Computer Science from Amherst College.

References

- [1] FFIEC, "FFIEC Issues Guidance on Authentication and Access to Financial Institution Services and Systems," Ffiec.gov, February 20, 2025. <http://www.ffiec.gov/news/press-releases/2021/pr-08-11>.
- [2] H. Flanagan (Editor), "Terminology in the IDPro Body of Knowledge," IDPro Body of Knowledge, vol. 1, no. 7, Sept. 2021, doi: 10.55621/idpro.41.
- [3] P. A. Grassi et al., National Institute of Standards and Technology, "Digital identity guidelines: authentication and lifecycle management," NIST Special Publication 800-63B, June 2017, doi: 10.6028/nist.sp.800-63b.

- [4] P. A. Grassi et al., National Institute of Standards and Technology, "Digital identity guidelines: federation and assertions," June 2017, doi: 10.6028/nist.sp.800-63c.
- [5] Natsuhiko Sakimura, John Bradley, Michael B. Jones, Breno de Medeiros, and Chuck Mortimore, "Final: OpenID Connect Core 1.0 incorporating errata set 1," openid.net. https://openid.net/specs/openid-connect-core-1_0.html
- [6] D. Hardt, "The OAuth 2.0 Authorisation Framework," Oct. 2012, doi: 10.17487/rfc6749.
- [7] Gaithersburg, MD: U.S. Department of Commerce, Joint Task Force, National Institute of Standards and Technology, "Security and Privacy Controls for Information Systems and Organisations," Security and Privacy Controls for Information Systems and Organisations, vol. 5, no. 5, Sept. 2020, doi: 10.6028/nist.sp.800-53r5.
- [8] M. Jones, J. Bradley, and N. Sakimura, "JSON Web Token (JWT)," May 2015, doi: 10.17487/rfc7519.
- [9] S. Jain, M. Mandell Freeman, B. Biggio, M. Duermeth, and G. Giacinto, "Who Are You? A Statistical Approach to Measuring User Authenticity," Jan. 2016, doi: 10.14722/ndss.2016.23240.
- [10] B. Campbell, J. Bradley, N. Sakimura, and T. Lodderstedt, "OAuth 2.0 Mutual-TLS Client Authentication and Certificate-Bound Access Tokens," Feb. 2020, doi: 10.17487/rfc8705.
- [11] David Freeman, Sakshi Jain, Markus Dürmuth, Battista Biggio, and Giorgio Giacinto, "Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0." [Online]. Available: <https://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf>
- [12] S. Cantor, J. Moreh, R. Philpott, E. Maler, "Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, March 15 2005.
- [13] World Wide Web Consortium, "Web Authentication: An API for accessing Public Key Credentials - Level 2," W3.org, April 08, 2021. <https://www.w3.org/TR/webauthn-2/>
- [14] F. Com, / CISA, C. Gov, and C. Gov, "CISA | DEFEND TODAY, SECURE TOMORROW Implementing Phishing-Resistant MFA," 2022. [Online]. Available: <https://www.cisa.gov/sites/default/files/2023-01/fact-sheet-implementing-phishing-resistant-mfa-508c.pdf>
- [15] F. Com, / CISA, C. Gov, and C. Gov, "CISA | DEFEND TODAY, SECURE TOMORROW Implementing Number Matching in MFA Applications," 2022. [Online]. Available: <https://www.cisa.gov/sites/default/files/publications/fact-sheet-implement-number-matching-in-mfa-applications-508c.pdf>
- [16] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero trust architecture," NIST Special Publication 800-207, vol. 1, no. 800-207, Aug. 2020, doi: 10.6028/nist.sp.800-207.
- [17] Lodderstedt, T., McGloin, M., & Hunt, P. (2013). OAuth 2.0 threat model and security considerations (RFC 6819). Internet Engineering Task Force. <https://doi.org/10.17487/RFC6819>
- [18] Szalachowski, P. (2020). Password-authenticated decentralized identities. arXiv. <https://doi.org/10.48550/arXiv.2007.15881>
- [19] P. A. Grassi, M. E. Garcia, and J. L. Fenton, Digital Identity Guidelines, NIST Special Publication 800-63-3, National Institute of Standards and Technology, Gaithersburg, MD, 2017. doi: 10.6028/NIST.SP.800-63-3.
- [20] K. Grizzle, M. Ansari, E. Wahlstroem, and C. Mortimore, "System for Cross-domain Identity Management: Protocol," Internet Engineering Task Force, September 2015, doi: 10.17487/rfc7644.