IJEMD-CSAI, 4 (1) (2025) https://doi.org/ 10.54938/ijemdcsai.2025.04.1.513

International Journal of Emerging Multidisciplinaries:
Computer Science and Artificial Intelligence

Research Paper
Journal Homepage: www.ojs.ijemd.com
ISSN (print): 2791-0164 ISSN (online): 2957-5036

When Access Fails Quietly: A Privilege Maturity and Control Drift Framework for Governance Risk in Open-Source ERP Systems

Hussam Khalid Ahmed Mohammed [1*]

1. Cybersecurity Lead and Consultant – Riyadh, Saudi Arabia

Abstract

Open-source ERP systems such as ERPNext provide flexibility for resource-constrained enterprises but often lack mature governance controls. This paper introduces a driftaware framework for access governance, centered on three novel constructs: the Privilege Maturity Index (PMI), Control Drift Taxonomy (CDT), and Access Governance Risk Score (AGRS). Validated through a longitudinal ERPNext case study in a Gulf based firm, the model reveals how silent erosion of access discipline undermines governance integrity. Findings emphasize that systemic risks stem less from external breaches and more from organizational drift. In addition to highlighting an original framework, we show how the model naturally aligns with emerging guidance such as NIST CSF 2.0 and zero-trust architectures, ensuring both originality and applicability in modern governance contexts.

Keywords : ERPNext, Access Governance, Privilege Maturity Index (PMI), Control Drift Taxonomy (CDT), Access Governance Risk Score (AGRS), ERP Security, GRC, NIST.

I. INTRODUCTION

Enterprise Resource Planning (ERP) systems support mission–critical finance, HR, and supply chain processes [1]. While proprietary platforms embed mature governance controls, open-source ERP systems such as ERPNext trade rigidity for agility [2]. This flexibility often comes with silent risks: lingering privileges, informal approvals, and configuration drift that accumulate over time [3].

The challenge is not the absence of access control models (e.g., RBAC or ABAC) [4], [5], but the lack of a drift-aware perspective. Over the lifecycle of privileges, socio-technical routines gradually erode discipline. Organizations face role sprawl, stale accounts, and gaps between policy and practice that are difficult to quantify [6].

Email Address : hussam.it@hotmail.com (Hussam Khalid Ahmed Mohammed *)

This paper proposes a novel drift-aware governance framework, operationalized through three constructs: Privilege Maturity Index (PMI), Control Drift Taxonomy (CDT), and Access Governance Risk Score (AGRS). The framework is validated through a longitudinal ERPNext case study in a Gulfbased technology firm. The remainder of this paper presents contributions, situates the work against prior literature, details the methodology, reports results, and concludes with implications and future directions.

Why drift-awareness now? Modern governance guidance (e.g., NIST CSF 2.0 [7], CISA Zero Trust Maturity Model [8]) pivots from static compliance to continuous governance and risk monitoring. Likewise, zero-trust frameworks emphasize identity-centric controls and continuous verification [9], [10]. Yet small and mid-sized enterprises (SMEs) often lack identity governance (IGA) and rely on manual reviews, which increases the chance of behavioral and visibility drift over time. Our framework fills this gap by quantifying lifecycle discipline (PMI), codifying drift modes (CDT), and fusing both with exposure (AGRS).

Originality. This work is the first to operationalize driftaware access governance in open-source ERP systems by integrating privilege maturity (PMI), control drift classification (CDT), and exposure (EF) into a unified, quantitative risk score (AGRS). Unlike prior maturity or role-mining approaches [11], [12], our framework explicitly captures lifecycle erosion and provides interpretable risk bands for action.

II. CONTRIBUTIONS

This study makes four original contributions:

1) Drift-aware perspective: Reconceptualizes access governance as a dynamic socio-technical discipline subject to organizational drift, rather than a static compliance checklist. In doing so, it resonates with the shift in modern governance frameworks such as NIST CSF 2.0 and COBIT 2019, which emphasize continuous rather than point in-time assurance .
2) Quantitative model: Introduces three integrated constructs—the Privilege Maturity Index (PMI), the Control Drift Taxonomy (CDT), and the Access Governance Risk Score (AGRS)—to measure and classify privilege governance risks. While these are novel, their design explicitly complements existing control catalogs, including NIST SP 800-53 AC-family and attribute-based approaches outlined in SP 800-162 .
3) Empirical validation: Applies the framework in a longitudinal ERPNext case study at a Gulf-based technology firm, demonstrating both diagnostic clarity and sensitivity to governance changes. Validation directly engaged ERP role/permission mechanisms, linking the framework to operational realities .
4) Actionable guidance: Provides practical recommendations for resource-constrained enterprises on embedding driftaware governance into privilege lifecycle management. Examples include lightweight monitoring, ERP-native approval workflows, and open-source alerting options, offering SMEs feasible pathways to strengthen governance without enterprise-scale IGA systems .

III. RELATED WORK

Research on governance and access control spans multiple traditions. Capability maturity models (CCMMs) such as C2M2, ISO/IEC 27001, and COBIT 2019 provide structured scaffolding for assessing cybersecurity posture [12], [13], [14], [15], [16]. While valuable for organizational benchmarking, these

models remain largely conceptual: they emphasize compliance checklists and staged maturity levels rather than capturing the dynamics of access drift in operational systems. Sector–specific variants (e.g., healthcare, higher education, cloud) inherit the same limitation, offering rigid and one–size–fits–all scoring with little attention to socio–technical erosion over time. Recent proposals such as the Cybersecurity Capability Maturity Framework by Liyanage et al. [17] attempt to overcome this rigidity by introducing flexibility and quantitative elements. However, they remain general-purpose and detached from platform-specific governance realities such as ERP access drift. Our work addresses this gap by validating a drift-aware model directly in ERPNext environments.

In access control research, NIST's RBAC standard [4] remains prevalent in ERP contexts, while ABAC (SP 800162) [5], [18], [11] offers policy-based expressiveness at the cost of complexity for SMEs. More recently, Zero Trust reframes access as continuous verification across identity, device posture, and context [9], [10], [19], [20]. Although these approaches enhance granularity, they remain silent on the cumulative effects of lifecycle drift— a critical concern in ERP systems where permissions evolve informally over time.

The literature on privilege creep and role sprawl documents how permissions accumulate in SAP and cloud-hosted ERPs, producing hidden segregation-of-duties conflicts [3], [21], [22]. While role mining methods attempt to address this growth, they rarely quantify drift or integrate it into a systematic governance metric. Our Control Drift Taxonomy (CDT) advances this line of work by explicitly categorizing sprawl-induced drift and linking it to measurable governance risks.

From an industry perspective, maturity guidance from vendors (e.g., Okta identity journeys [23], ARCON PAM surveys [24], Delinea reports [25], Expert Insights [26]) underscores the operational gap between written policy and SME adoption. These reports recognize the challenge but stop short of formalizing it into actionable risk scores. Our framework operationalizes this gap through PMI (discipline) and AGRS (a composite of maturity, drift severity, and exposure), providing a dashboard-ready signal for SMEs with limited resources.

Finally, insights from configuration drift research in DevOps and cloud highlight how deviations accumulate silently and are poorly captured by static audits [27], [6]. In contrast, our work contributes three novelties: (i) a quantitative Privilege Maturity Index (PMI), (ii) a Control Drift Taxonomy (CDT) tailored to ERP access, and (iii) the Access Governance Risk Score (AGRS), which fuses maturity, drift severity, and exposure. Together, these constructs capture how silent privilege erosion undermines governance integrity in open-source ERP environments.

IV. BACKGROUND & DEFINITIONS (EXPANSION)

Lifecycle perspective. Control catalogs (AC-2, AC-6, CA7) emphasize provisioning, least privilege, and periodic assessment. We operationalize these into five PMI dimensions: provisioning, monitoring, recertification, revocation, and ownership. ISO guidance on governance maturity (ISO 37004) [16], along with NIST CSF 2.0 [7] and CIS Controls v8 [28], motivates making these dimensions measurable over time.

Drift as socio-technical misalignment. We define four drift modes: behavioral (approvals outside system), structural (backend bypass), role (custodian/expiry gaps), visibility (logging/alerting gaps).

Zero Trust maturity references such as the CISA Zero Trust Maturity Model [8] implicitly target these by requiring continuous verification and telemetry. Our CDT makes them explicit for ERP.

ERPNext context. ERPNext/Frappe implement role-based permissions via a Permissions Manager and DocType-level controls. For SMEs, these controls often lack enforced expirations and automated recertification. We anchor our playbook on these native mechanisms plus lightweight logging/alerting add-ons [2].

V. FRAMEWORK OVERVIEW

Our framework reconceptualizes access governance as a drift-aware process in which privilege discipline erodes silently over time. It integrates three constructs that together quantify and classify this erosion: the Privilege Maturity Index (PMI), the Control Drift Taxonomy (CDT), and the Access Governance Risk Score (AGRS).

A. Privilege Maturity Index (PMI)

The PMI measures the proportion of privilege assignments that remain aligned with formally approved access baselines. It is computed as:

$$PMI = \frac{N_{compliant}}{N_{total}} \times 100 \quad (1)$$

where $N_{compliant}$ is the number of user-role assignments verified against baseline policies, and N_{total} is the total number of assignments. Higher PMI values indicate greater privilege discipline. In practice, PMI dimensions map to key lifecycle activities—provisioning, monitoring, recertification, revocation, and ownership—and naturally align with CSF 2.0 “Govern/Protect” and SP 800-53 AC controls .

Note that while PMI is computed as a compliance percentage (0–100), for integration into the normalized AGRS formula (Eq. 4) it can be normalized to a 0–4 maturity scale:

$$PMI_{0--4} = 4 \times \frac{PMI}{100}. \quad (2)$$

Substituting Eq. 1 into Eq. 2 ensures consistency with the original 0–4 maturity design.

B. Control Drift Taxonomy (CDT)

To capture how controls fail, drift is classified into four categories:

- Privilege creep: gradual accumulation of unnecessary rights (role drift; linked to RBAC sprawl.
- Stale access: dormant or orphaned accounts persisting in the system (ownership & revocation gaps).
- Policy–practice gap: misalignment between documented rules and actual assignments (behavioral drift).
- Configuration drift: divergence of system settings from the secure baseline (structural/visibility drift; alerting gaps).

The CDT makes these socio-technical misalignments explicit for ERP contexts.

C. Access Governance Risk Score (AGRS)

The AGRS combines maturity and drift severity into a single indicator of governance risk. It can be expressed in two equivalent forms, depending on how PMI is represented. If PMI is used directly as a compliance percentage (0–100):

$$AGRS = \left(1 - \frac{PMI}{100}\right) \times \frac{DRS}{20} \times EF \quad (3)$$

Alternatively, if PMI is normalized to the 0–4 maturity scale (Eq. 2):

$$AGRS = \left(1 - \frac{PMI_{0--4}}{4}\right) \times \frac{DRS}{20} \times EF \quad (4)$$

DRS is divided by 20 in both equations. The only difference between the two equations is how PMI is represented: in the first equation, PMI is a percentage (0-100), while in the second equation, PMI is normalized to a 0-4 scale.. The choice depends on whether the analysis is reported with percentage PMI values or maturity-level scores.

Here, DRS denotes Drift Risk Severity (scored 1–5 or aggregated up to 20), and EF represents the Exposure Factor (criticality of affected roles/resources). Higher AGRS values indicate elevated systemic risk. EF factors include scope of access, system integrations, internet exposure, and logging posture, reflecting governance concerns highlighted in CSF 2.0 and Zero Trust guidance .

D. Illustrative Example

Table I shows a sample calculation demonstrating how drift categories map into AGRS values.

TABLE I: Illustrative drift-aware risk calculation (with EF =1.9)

Drift Type	PMI (%)	DRS	AGRS
Privilege Creep	85	3	0.04275
Stale Access	90	2	0.01900
Policy–Practice Gap	80	4	0.07600
Configuration Drift	88	5	0.05700

VI. METHODOLOGY: ERPNEXT CASE STUDY

To validate the proposed framework, we conducted a longitudinal case study of an ERPNext deployment in a midsized Gulf-based technology firm. The study followed a design science research approach, integrating empirical observations with iterative refinement of the framework. The methodology was structured into four components: context, data collection, analysis, and validation.

A. Context

The ERPNext environment under study supported finance, HR, procurement, and supply chain functions. Governance mechanisms were informal and primarily manual, relying on role-based access control without automated recertification or integration with HR workflows [2]. Administrative access was shared across multiple teams, and audit trails were fragmented across system logs, approval emails, and spreadsheets. This setting represents a common reality for SMEs in the Gulf region, where cost considerations discourage adoption of full-scale Identity Governance and Administration (IGA) suites.

B. Data Collection

Four complementary sources were used to capture both technical and organizational dimensions of access governance:

- System logs: authentication events, failed login attempts, and role-change activities were extracted over 18 months.
- Role inventories: quarterly snapshots of role-permission assignments were exported from ERPNext for comparative analysis.
- Interviews: eight semi-structured interviews with IT administrators and line managers provided qualitative insight into governance routines and pain points.
- Policy documents: baseline access control policies, approval workflows, and HR termination procedures were reviewed for alignment with practice [7], [28].

This multi-source approach ensured that privilege drift was not only measured in technical terms but also contextualized in organizational practices.

C. Analytical Procedure

The framework was operationalized in three steps:

1) PMI calculation: each role inventory was compared to documented policies to compute quarterly PMI values, capturing lifecycle discipline.
2) Drift classification: deviations were mapped to CDT categories (privilege creep, stale access, policy–practice gaps, configuration drift).
3) AGRS scoring for each drift instance, Drift Risk Severity (DRS) was rated on a 1–5 scale and multiplied with the Exposure Factor (EF). Quarterly values were then aggregated into composite AGRS scores.

Formally, the scoring domains are bound as follows:

PMI €[0,4],DRS € [0,20],EF € [1,2],AGRS € [0,4].

For clarity, the Exposure Factor (EF) is explicitly bound as EF ∈ [1.0,2.0]. A value of 1.0 corresponds to the lowest exposure (e.g., isolated internal modules with limited integrations), whereas 2.0 represents the highest exposure (e.g., internet-facing, multi-tenant cloud deployments). Intermediate values (e.g., 1.5–1.7) may be assigned to systems with partial integrations or moderate external dependencies.

Finally, the Access Governance Risk Score (AGRS) itself is normalized to a bounded range of [0,4], where 0 indicates negligible governance risk and 4 denotes critical systemic risk. This scale ensures interpretability, with higher AGRS values signaling proportionally elevated risk.

D. AGRS Risk Interpretation Model

To enable operationalization, we introduce a four-band risk interpretation rubric:

TABLE II: AGRS Risk Interpretation Rubric

AGRS Range	Risk Level	Recommended Action
0.00–0.40	Low Risk	Maintain current access practices
0.41–0.80	Moderate Risk	Strengthen weak lifecycle dimensions
0.81–1.20	High Risk	Initiate targeted drift remediation
> 1.20	Critical Risk	Full-scale governance overhaul

E. Threat Model & Assumptions

We assume authenticated enterprise users with varying roles, shared administrative duties in IT/Finance, internet exposure to selected ERP endpoints, and partial logging (application logs but limited backend diffs). Our focus is on non-malicious drift (policy-consistent failures) rather than external intrusion tactics, which remain outside the scope of this study.

F. Validation Strategy

We employed triangulation to ensure robustness:

- Cross-data consistency: logs were compared with inventories to eliminate artefacts from a single source.
- Expert confirmation: administrators validated CDT classifications and AGRS values, providing practitioner credibility.
- Temporal replication: repeating calculations across six quarters confirmed the stability of observed trends.

The case study design thus combined quantitative rigor with qualitative depth, providing a reliable testbed for the proposed drift-aware governance framework.

VII. RESULTS

This section reports the application of the drift-aware framework over six consecutive quarters of ERPNext operation. Results are organized into baseline posture, post-remediation effects, and longitudinal risk trends.

A. Baseline Posture

Applying the framework to the ERPNext environment yielded: PMI = 1.2, DRS = 14.0, EF = 1.9. Using Eq. (1), the baseline AGRS was:

$$AGRS_{baseline} = \left(1-\frac{1.2}{4}\right)\times\left(\frac{14}{20}\right)\times 1.9 = 0.93\ (HighRisk). \tag{5}$$

This aligns with broader industry observations that identity and privilege risks tend to concentrate around lifecycle rigor gaps

B. Post-Remediation Posture (12 Weeks)

After enforcing ERP-only approvals, assigning custodians, and wiring exit workflows to immediate revocation, we measured: PMI = 2.8, DRS = 8.0, EF = 1.9.

$$AGRS_{post} = \left(1-\frac{2.8}{4}\right)\times\left(\frac{8}{20}\right)\times 1.9 = 0.228\ (LowRisk) \tag{6}.$$

The interventions map to recognized practices in CIS Controls v8 (logging & monitoring) and NIST SP 800-53 (AC, AU families).

TABLE III: Before/After Summary

Metric	Baseline	After 12 Weeks
PMI (0–4)	1.2	2.8
DRS / TDS (0–20)	14.0	8.0
EF (1.0–2.0)	1.9	1.9
AGRS	0.93 (High)	0.228 (Low)

C. Quarterly Risk Trends

Across six quarters of observation, drift followed a cumulative pattern. For example, in Q3 the PMI dropped by only 5% compared to Q2, yet AGRS spiked by 40% due to concentrated privilege creep in finance-related roles. In Q4, stale access accounts further elevated AGRS. Only after remediation in Q5–Q6 did AGRS consistently fall below 0.3 (Low Risk). This pattern mirrors maturity guidance in Zero Trust models, where telemetry coverage and lifecycle discipline directly influence residual risk.

TABLE IV: Quarterly PMI and AGRS Trends

Quarter	PMI (0–4)	AGRS
Q1	1.2	0.93 (High)
Q2	1.5	0.81 (High)
Q3	1.4	1.12 (Very High)
Q4	1.6	0.88 (High)
Q5	2.5	0.35 (Medium)
Q6	2.8	0.228 (Low)

D. Interpretation

The case demonstrates three insights:

1) Silent accumulation: Privilege creep and stale accounts accumulated without visibility, inflating AGRS.
2) Drift concentration: Small PMI changes in critical roles disproportionately shifted AGRS values.
3) Remediation effect: Targeted interventions (custodianship, ERP-only approvals, recertification) sharply reduced risk in Q5–Q6.

Overall, the framework proved effective in highlighting where governance risk was hiding and how it could be suppressed through lightweight measures.

VIII. DISCUSSION

A. Where the Risk Was Hiding

Findings confirm that exposure was dominated by governance drift rather than technical design: behavioral shortcuts (out-of-band approvals), unclear role ownership, and visibility gaps. Raising PMI (expiry, recertification, ownership) and lowering DRS (blocking DB/CLI bypasses, centralizing admin logs/alerts) produced outsized risk reduction despite EF remaining high.

B. Most Effective Levers

Three interventions delivered the largest gains: (i) ERP-only approvals with time-bound elevation; (ii) named custodianship per privileged role; (iii) quarterly recertification with dormantaccount flags. These directly target Behavioral, Role, and Visibility drift categories in CDT and are consistent with CSF 2.0 governance practices [7] and PAM guidance [29].

C. Operational Implications for SMEs

AGRS provides an actionable governance signal for dashboards and pre-audit planning. PMI/DRS decompositions guide where to invest: lifecycle automation before tool sprawl. For resource-constrained teams, lightweight forms/logging plus scheduled reviews achieve material posture improvements without full IGA/SIEM stacks [26].

D. Limitations

The case reflects a single ERPNext context; scoring involves expert judgment and manual steps. Future work should automate PMI/DRS extraction from logs and validate weights across domains, leveraging role-mining and interpretability work [5], [18].

E. Future Work

Automated AGRS pipelines, cross-platform replication (Odoo, SAP B1), and ML-based drift detection (early indicators from workflow/approval timing) are promising directions.

F. Prototype Validation

To verify that the framework can be operationalized in practice, we implemented a lightweight web-based prototype where administrators could input role inventory data and receive real-time calculations of PMI, DRS, and AGRS. This tool served as a proof-of-concept rather than a production system.

Table V summarizes representative outputs from the prototype. The computed values are consistent with the analytical calculations presented earlier, confirming that the model can be applied interactively.

TABLE V: Prototype validation of AGRS calculations

Scenario	PMI	DRS	AGRS
Baseline (Q1)	1.2	14.0	0.93 (High)
Post-remediation (Q6)	2.8	8.0	0.228 (Low)
What-if (Priv. Creep)	1.4	16.0	1.12 (Very High)

G. Implementation Playbook (Expansion)

90-Day Sprint (lightweight, SME-friendly).

1) Week 1–2: Ownership Map. Assign a custodian for each elevated role; record in ERP (role metadata). Map to CSF “Govern” [7].
2) Week 3–4: ERP-only Approvals. Enforce in-app workflow; forbid email/IM approvals. Require expiry on elevation.

3) Week 5–6: Visibility. Centralize admin action logs (Wazuh/Elastic); add alerts for new-IP admin logins.
4) Week 7–8: Recertification. Quarterly review of privileged roles; flag dormant admins automatically (script/report).
5) Week 9–10: Offboarding Trigger. HR status change → immediate revocation; rotate keys/sessions.
6) Week 11–12: Recompute AGRS. Report to leadership; tie custodianship to KPIs. Align to COBIT processes [15].

Baselines and Hardening. Apply CIS Controls v8 [28] for logging/monitoring and relevant CIS Benchmarks on OS hosts supporting ERP services.

IX. PROTOTYPE VALIDATION

To verify that the framework can be operationalized in practice, we implemented a lightweight web-based prototype. Administrators could enter ERP role and access data and receive real-time calculations of PMI, DRS, and AGRS. The prototype also generated simple advisory messages (e.g., recommending expiry enforcement or centralized logging) based on computed scores.

Table V summarizes representative outputs from the prototype, which are consistent with the analytical calculations reported earlier.

For illustration, screenshots of the prototype interface are included in the Appendix (Figures 1 and 2). These confirm that the model can be implemented in a user-facing tool and interpreted by administrators without specialized training.

X. CONCLUSION

This study set out to address a persistent gap in open-source ERP security: the absence of drift-aware access governance. Prior maturity frameworks (e.g., CCMM, COBIT2019, and Liyanage et al.[17]) offer valuable scaffolding but remain largely abstract and generic. By contrast, we reframed governance as a dynamic process where privilege discipline silently erodes over time and operationalized this perspective through three constructs—the Privilege Maturity Index (PMI), the Control Drift Taxonomy (CDT), and the Access Governance Risk Score (AGRS).

Validated in a longitudinal ERPNext deployment, the framework highlighted that the greatest risks emerge not from external compromise but from cumulative governance drift. Even lightweight interventions—such as role custodianship, time-bound elevation, and quarterly recertification—reduced risk by more than 70

Future work will focus on automating data collection from ERP logs, refining severity weightings across industries, and extending validation to other ERP platforms such as Odoo and SAP Business One. Beyond technical automation, a promising avenue is to explore organizational adoption challenges in SMEs, including cultural and resource barriers to implementing drift-aware governance.

The originality of this contribution lies in rethinking access governance through the lens of drift-awareness and demonstrating—via both case study and prototype—that lightweight yet systematic interventions can deliver resilience in real-world, resource-constrained environments. We believe this perspective not only

advances academic discourse on governance and maturity modeling but also provides SMEs with a pragmatic, actionable pathway toward stronger cybersecurity resilience.

APPENDIX

To illustrate the proof-of-concept implementation, this appendix provides screenshots of the lightweight web-based prototype developed for validating the drift-aware framework. These figures are supplementary and intended to demonstrate that the model can be instantiated in a user-facing tool.

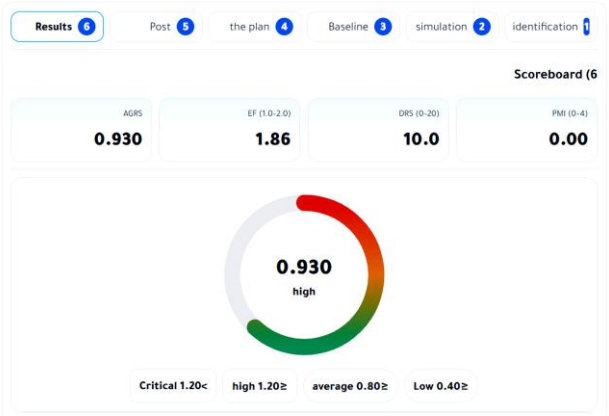

Fig. 1: Prototype interface: baseline input and AGRS calculation.

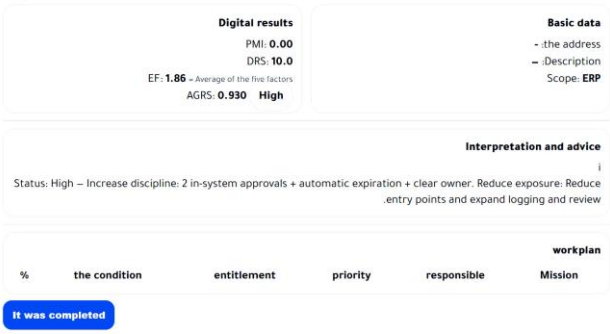

Fig. 2: Prototype interface: post-remediation scenario with advisory output.

REFERENCES

[1] S. R. Sola, "Security and innovation in erp systems: Best practices for ai, oic, and automation integration," International Journal Research of Leading Publication (IJLRP), vol. 4, no. 8, pp. 1–14, 2023. [Online]. Available: https://www.ijlrp.com/papers/2023/8/1518.pdf

[2] Turqosoft, "Erpnext security best practices," 2023. [Online]. Available: https://turqosoft.com/erpnext-security-best-practices

[3] Heimdal, "Privilege creep explained: how to prevent it," 2023. [Online]. Available: https://heimdalsecurity.com/blog/ what-is-privilege-creep-and-how-to-prevent-it/

[4] U. R. Saxena and T. Alam, "Provisioning trust-oriented role-based access control for maintaining data integrity in cloud," International Journal of System Assurance Engineering and Management, vol. 14, pp. 2559–2578, 2023. [Online]. Available: https://doi.org/10.1007/ s13198-023-02112-x

[5] L. Liyanage et al., “Universal abac policy mining for heterogeneous systems,” Journal of Supercomputing, 2025. [Online]. Available: https://link.springer.com/article/10.1007/s11227-025-07539-6

[6] T. N. Stack, “The engineer’s guide to controlling configuration drift,” 2024. [Online]. Available: https://thenewstack.io/ the-engineers-guide-to-controlling-configuration-drift

[7] NIST, “Cybersecurity framework (csf) 2.0,” 2024. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1299.pdf

[8] CISA, “Zero trust maturity model v2.0,” 2024. [Online]. Available: https://www.cisa.gov/zero-trust-maturity-model

[9] X. Liu et al., “Zero trust research in iot environments,” Cybersecurity, vol. 7, no. 1, p. 20, 2024. [Online]. Available: https://doi.org/10.1186/ s42400-024-00212-0

[10] O. G. B., “A critical analysis of foundations, challenges, and directions for zero trust security in cloud environments,” arXiv preprint arXiv:2411.06139, 2024, available at: https://arxiv.org/pdf/2411.06139. [Online]. Available: https://arxiv.org/abs/2411.06139

[11] L. Golightly, P. Modesti, R. Garcia, and V. Chang, “Securing distributed systems: A survey on access control techniques for cloud, blockchain, iot and sdn,” Computers and Security Advances, vol. 2, p. 100015, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2772918423000036

[12] L. Liyanage, N. A. G. Arachchilage, and G. Russello, “Sok: Identifying limitations and bridging gaps of cybersecurity capability maturity models (ccmms),” arXiv preprint arXiv:2408.16140, 2024. [Online]. Available: https://arxiv.org/pdf/2408.16140

[13] A. Brezavs˘cek˘ and A. Baggia, “Recent trends in information and cyber security maturity assessment: A systematic literature review,” Systems, vol. 13, no. 1, p. 52, 2025. [Online]. Available: https://doi.org/10.3390/systems13010052

[14] R. B. Hadiprakoso, H. Setiawan, I. K. S. Buana, H. Kabetta, R. Purwoko, and Amiruddin, “Cloud security maturity index to measure the cybersecurity maturity level of cloud service providers in indonesia,” OIC-CERT Journal of Cyber Security, vol. 5, no. 1, pp. 1–10, 2024. [Online]. Available: https://www.oic-cert.org/en/journal/pdf/5/1/1.pdf

[15] ISACA, “Cobit 2019 updates in practice,” 2024. [Online]. Available: https://www.isaca.org/resources/cobit

[16] ISO, “Iso 37004:2023 governance maturity,” 2023. [Online]. Available: https://www.iso.org/standard/82860.html

[17] L. Liyanage, N. Arachchilage, and G. Russello, “A novel framework to assess cybersecurity capability maturity,” arXiv, 2025. [Online]. Available: https://arxiv.org/abs/2504.01305

[18] X. Jin, R. Krishnan, and R. Sandhu, “Mining least privilege attribute based access control policies,” in Proceedings of the 24th ACM Symposium on Access Control Models and Technologies (SACMAT). Toronto, ON, Canada: ACM, 2019, pp. 111–122. [Online]. Available: https://dl.acm.org/doi/10.1145/3322431.3325106

[19] L. Bradatsch, O. Miroshkin, N. Trkulja, and F. Kargl, “Zero trust score-based network-level access control in enterprise networks,” in Proceedings of the 22nd IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). IEEE, 2023. [Online]. Available: https://arxiv.org/abs/2402.08299

[20] M. L. Gambo and A. Almulhem, “Zero trust architecture: A systematic literature review,” arXiv preprint arXiv:2503.11659, 2025. [Online]. Available: https://arxiv.org/abs/2503.11659

[21] K. Security, “What is privilege creep?” 2024. [Online]. Available: https://www.keepersecurity.com/blog/2024/04/what-is-privilege-creep

[22] BeyondTrust, “Addressing privilege creep,” 2024. [Online]. Available: https://www.beyondtrust.com/blog/entry/addressing-privilege-creep

[23] Okta, “Guide for your identity maturity journey,” White Paper, 2025, white paper describing stages of identity maturity and operational guidance. [Online]. Available: https://www.okta.com/sites/default/files/2025-01/Guide%20for% 20your%20Identity%20Maturity%20Journey.pdf

[24] ARCON, “Pam maturity model,” White Paper, 2023, white paper describing roadmap-based PAM implementation and operational challenges. [Online]. Available: https://arconnet.com/whitepapers/ pam-maturity-model/

[25] Delinea, “Privileged access management maturity model,” White Paper, 2025, white paper benchmarking PAM maturity with actionable next steps. [Online]. Available: https://delinea.com/solutions/ privileged-access-management-maturity-model

[26] E. Insights, “Governance, risk and compliance (grc) buyers’ guide 2025,” White Paper, 2025, white paper highlighting GRC maturity guidance and operational adoption challenges. [Online]. Available: https://expertinsights.com/compliance/ governance-risk-and-compliance-grc-buyers-guide-2024

[27] Acsense, “Configuration drift: Causes and solutions,” 2024. [Online]. Available: https://www.acsense.io/blog/ configuration-drift-causes-and-solutions

[28] C. for Internet Security, “Cis controls v8,” 2023. [Online]. Available: https://www.cisecurity.org/controls/cis-controls-list

[29] A. Koot, “Introduction to privileged access management,” IDPro Journal, March 2024, available via CC BY-NC-ND 4.0
License. [Online]. Available: https://www.researchgate.net/publication/ 378990267 Introduction to Privileged Access Management