Session-Level Threat Scoring for Privileged Operations in Elastic Infrastructure

Authors

Oliver J. Bennett[1], Amelia R. Clarke[2*]

Affiliations

School of Computing and Information Systems, Faculty of Engineering and Information Technology, The University of Melbourne, Parkville, VIC 3010, Australia

***Corresponding author:** a.clarke@unsw.edu.au

Abstract

Privileged access sessions are widely used in cloud infrastructure for system administration, database maintenance, incident response, and production troubleshooting. However, vulnerable bastion hosts, weak session controls, excessive administrator privileges, and incomplete command auditing may create serious data-security risks. This study proposes a policy-driven risk quantification framework for vulnerable privileged access sessions in large-scale cloud environments. The framework combines session privilege level, target asset sensitivity, bastion-host vulnerability status, command-audit completeness, session duration, just-in-time access compliance, and policy violation records into a unified risk score. Policy constraints are derived from privileged access management requirements, session recording rules, administrator separation policies, database access controls, and emergency-access approval standards. Experiments are conducted on a cloud operation environment containing 620 bastion hosts, 18,400 administrator identities, 76,000 privileged session records, 9,300 production assets, 2,870 sensitive databases, and 13,600 vulnerability findings from bastion systems, remote access agents, and management interfaces. The proposed framework identifies 1,360 high-risk privileged-access chains, including vulnerable bastion hosts connected to sensitive databases, long-lived emergency sessions, administrators with cross-environment privileges, and sessions lacking complete command logs. It consolidates 13,600 vulnerability findings into 1,890 privileged-access remediation cases. The median risk-scoring time is 33 ms per session, and the full privileged-access assessment completes in 9.7 minutes. After one remediation batch, high-risk database administration paths decrease from 486 to 173, while incomplete session-audit cases drop by 2,140 records. Risk-score drift remains below 0.16 points on a ten-point scale across four access-policy snapshots. The results show that policy-driven risk quantification can help prioritize privileged-access vulnerabilities that directly affect cloud data confidentiality and operational accountability.

Keywords: Privileged access management; bastion host security; cloud vulnerability risk quantification; policy-driven risk scoring; administrator session audit; cloud data security; access governance.

1. Introduction

Privileged access is a necessary function in cloud infrastructure, but it also introduces a concentrated source of security risk. Cloud administrators, database operators, incident-response engineers, and production support staff often rely on elevated permissions to maintain systems, repair faults, change configurations, and handle production incidents [1,2]. These privileged sessions usually pass through bastion hosts, remote access agents, privileged access management platforms, database consoles, and cloud management interfaces. Once a privileged session is misused or compromised, attackers may gain access to production assets, sensitive databases, cloud credentials, and audit-sensitive operations. Recent work on open-source cloud infrastructure security has shown that vulnerability information, deployment context, policy checks, and remediation logic can be integrated into a practical security framework for cloud environments [3]. Related studies on cloud security, zero-trust architecture, and access governance further emphasize that identity, access context, device state, workload status, and data sensitivity should be assessed together when protecting cloud resources [4,5].

Privileged access is different from ordinary user access because it directly connects human operators, management interfaces, production workloads, and sensitive data assets. A short emergency session may modify security groups, restart critical services, export database records, rotate credentials, or change logging policies [6,7]. Therefore, the risk of privileged access cannot be judged only by whether authentication succeeds or whether an approval ticket exists. The actual risk depends on who initiates the session, which access path is used, whether the bastion host is vulnerable, what commands are executed, how completely the commands are audited, how long the session lasts, and whether the target asset contains regulated or sensitive data. Research on privileged behavior auditing has stressed the need to record administrator operations and detect abnormal activities in cloud platforms [8,9]. However, many existing studies still treat access approval, authentication, audit logging, trust evaluation, and vulnerability management as separate problems. They rarely quantify the risk of a real privileged session by jointly considering session privilege level, target asset sensitivity, bastion-host vulnerability, command-audit completeness, session duration, emergency access approval, and policy violations [10,11]. Vulnerability management has also moved from static severity scoring toward risk-based prioritization. Recent studies have used exploit prediction, offensive-security evidence, CVSS, EPSS, KEV records, asset context, and machine-learning methods to improve vulnerability ranking [12,13]. These methods show that vulnerability risk should be evaluated in its operating context rather than by a technical severity score alone [14,15]. This point is especially important for privileged access scenarios. A medium-severity weakness on a bastion host may become a serious operational risk when the host is used by administrators with cross-environment privileges to access restricted databases. In contrast, a high-severity vulnerability may have limited practical impact if the

affected system is isolated, has no privileged access path, and does not connect to sensitive assets [16,17]. This means that vulnerability risk in privileged access chains is shaped by the interaction between technical weakness, access privilege, target sensitivity, and operational policy [18,19]. Recent privilege-analysis studies have further shown that over-privileged identities, complex cloud roles, and hidden access-path relations can create serious risks in multi-cloud and large-scale cloud systems [20,21]. These studies are useful for detecting excessive permissions, privilege escalation paths, and risky identity relationships. However, they seldom provide a session-level risk score that links administrator behavior, access path, bastion vulnerability, target asset sensitivity, and remediation priority. In practical cloud operations, security teams often receive separate findings from vulnerability scanners, identity governance tools, bastion logs, privileged access management systems, and compliance platforms. These fragmented findings make it difficult to determine which privileged-access chains should be fixed first. As a result, remediation may focus on isolated high-CVSS vulnerabilities while overlooking access paths that create greater exposure to sensitive data and critical production operations [22,23].

To address this gap, this study proposes a policy-driven risk quantification method for vulnerable privileged access sessions in cloud infrastructure. The method treats the privileged-access chain as the main analysis unit and links administrator identity, access path, bastion host, target asset, vulnerability finding, command-audit status, session duration, just-in-time access compliance, and policy violation records. It combines session privilege, target sensitivity, bastion vulnerability status, command-audit completeness, session duration, emergency access approval, and policy impact into one risk score. The purpose is not only to rank vulnerable hosts, but also to identify privileged sessions that may create high downstream exposure to sensitive data, production services, and audit-critical operations. The proposed method is evaluated in a cloud operation environment containing 620 bastion hosts, 18,400 administrator identities, 76,000 privileged session records, 9,300 production assets, 2,870 sensitive databases, and 13,600 vulnerability findings. By integrating vulnerability context, privileged-access paths, target-asset sensitivity, and policy status, the method groups raw findings into practical remediation cases. This design helps security teams move from isolated vulnerability lists to access-chain-level risk management. The expected contribution of this study is to provide a more operationally useful basis for prioritizing fixes, reducing privileged-access exposure, protecting cloud data confidentiality, and improving accountability in production cloud environments.

2. Materials and Methods

2.1 Sample and Study Environment Description

This study was performed in a large cloud operation environment involving system administration, database maintenance, incident response, and production troubleshooting. The dataset contained 620 bastion hosts, 18,400 administrator identities, 76,000 privileged session records, 9,300 production assets, 2,870 sensitive databases, and 13,600 vulnerability findings. The

vulnerability findings were obtained from bastion systems, remote access agents, management interfaces, and access-control components. Each privileged session was used as the basic unit of analysis. For each session, the administrator identity, access role, bastion host, target asset, session duration, approval status, command-audit status, just-in-time access status, and policy violation record were collected. A session was included only when it had a valid administrator identity, a clear access path, a known target asset, and complete policy metadata. Sessions with missing identity mapping, incomplete target records, invalid timestamps, or repeated session identifiers were removed before analysis.

2.2 Experimental Design and Control Settings

The experiment compared the policy-driven session-level risk model with three common assessment methods. The experimental group used the session-level model. In this model, privileged access risk was measured by combining session privilege level, target asset sensitivity, bastion-host vulnerability status, command-audit completeness, session duration, just-in-time access compliance, and policy violation records. Three control groups were used. The first control group ranked sessions only by vulnerability severity on bastion hosts and remote access agents. The second control group used asset sensitivity only, in which sessions linked to sensitive databases and production assets were assigned higher priority. The third control group used policy violation count, in which sessions were ranked by the number of failed access-policy checks. These control settings represent common practices in vulnerability management, data asset protection, and access governance. The comparison was used to test whether the session-level model could identify higher-risk privileged-access chains than severity-only, asset-only, or rule-count-based methods.

2.3 Measurement Methods and Quality Control

Security and access data were collected from privileged access management records, bastion-host logs, session recording systems, vulnerability scanning reports, cloud identity records, database access logs, approval workflows, and policy audit reports. Vulnerability findings were standardized by host identifier, vulnerability ID, affected component, severity level, exploitability label, and discovery time. Privileged session records were standardized by administrator identity, role type, session start time, session end time, target asset, access protocol, approval source, and command-audit status. Target asset sensitivity was divided into four levels: low, medium, high, and critical. Command-audit completeness was measured by checking login records, command records, file operation records, and session termination records. Just-in-time compliance was measured by comparing session time, approval window, granted privilege, and target scope. Quality control was performed in three steps. First, repeated vulnerability findings were removed according to host identifier, vulnerability ID, and component version. Second, inconsistent access records were checked against bastion logs and cloud identity snapshots. Third, session-chain

completeness was verified by requiring each retained record to contain a valid administrator identity, access path, bastion host, and target asset. A random 5% subset of privileged sessions was manually reviewed to confirm the accuracy of session mapping and policy labeling.

2.4 Data Processing and Model Formulation

All raw records were converted into privileged-session risk units before model calculation. Each vulnerability finding was mapped to related sessions according to the affected bastion host, remote access agent, management interface, or access-control component. Numerical variables, including session duration, privilege breadth, vulnerability severity, and policy violation count, were normalized to a range from 0 to 1. Categorical variables, including target asset sensitivity, command-audit status, just-in-time compliance, and administrator role type, were encoded as binary or ordinal values according to their security meaning. The risk score of privileged session iii was calculated as follows:

$$R_i=\alpha V_i+\beta S_i+\gamma P_i+\delta A_i+\lambda D_i+\mu J_i$$

where R_i is the final privileged-session risk score. V_i is the normalized bastion-host vulnerability score, S_i is the target asset sensitivity score, P_i is the session privilege score, A_i is the command-audit incompleteness score, D_i is the normalized session duration score, and J_i is the just-in-time non-compliance score. The weights $\alpha,\beta,\gamma,\delta,\lambda,$and μ were set according to security impact and adjusted through sensitivity analysis. The remediation reduction ratio was calculated as follows:

$$RR=Hbefore−HafterHbefore×100\%$$

Where H_{before} and H_{after} are the numbers of high-risk privileged-access chains before and after remediation, respectively. This metric was used to measure whether prioritized fixes reduced more risky administrator access paths than the control methods.

2.5 Model Evaluation and Validation Procedure

Model performance was evaluated using high-risk chain identification, remediation priority, scoring efficiency, and score stability. High-risk chain identification was measured by the number and type of privileged-access paths detected by each method. These paths included vulnerable bastion-to-database paths, long-lived emergency sessions, cross-environment administrator sessions, and sessions with incomplete command logs. Remediation priority was evaluated by applying the top-ranked remediation cases and comparing the reduction in high-risk database administration paths and incomplete session-audit records. Scoring efficiency was measured by the median risk-scoring time per session and the total assessment time for all privileged session records. Score stability was tested across four access-policy snapshots to determine whether the model remained stable when policy rules and access relationships changed. To reduce the effect of a single remediation order, the prioritized remediation list was tested under several ranking windows. The results were compared with the three control groups to determine whether the

policy-driven session-level model provided clearer remediation value and better practical use for cloud privileged access security.

3. Results and Discussion

3.1 Identification of High-Risk Privileged-Access Chains

The policy-driven model identified 1,360 high-risk privileged-access chains from 76,000 privileged session records and 13,600 vulnerability findings. These chains were mainly related to vulnerable bastion hosts connected to sensitive databases, long-lived emergency sessions, administrators with cross-environment privileges, and sessions with incomplete command logs. The results show that privileged access risk was not determined by vulnerability severity alone. Some high-severity bastion-host findings had limited impact because the related sessions accessed low-sensitivity assets or had complete audit records. In contrast, several medium-severity findings became high-risk cases when they were linked to sensitive databases, broad administrator privileges, weak session controls, or missing command logs. This result is consistent with previous studies on privileged user auditing, which report that cloud platforms may face serious risks when privileged accounts are leaked or misused [24,25]. However, most existing methods focus on behavior detection and post-event audit evidence. The present study further links vulnerability status, session behavior, policy compliance, and target asset sensitivity into one privileged-access chain, as shown in Fig. 1.

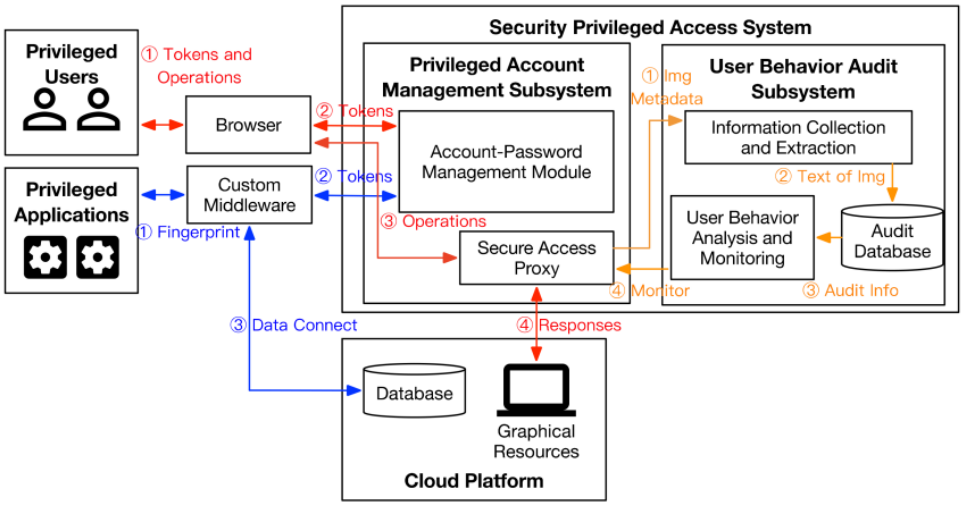

Fig. 1. Structure of privileged-access chains involving bastion hosts, administrator sessions, target assets, and command-audit records.

3.2 Remediation Case Consolidation and Practical Effect

The model consolidated 13,600 vulnerability findings into 1,890 privileged-access remediation cases. This result shows that many raw findings were repeated across bastion hosts, remote access agents, and management interfaces that supported the same access path [26,27]. If each finding is handled as an independent task, the remediation list becomes long and difficult to

manage. The proposed method reduces this problem by grouping findings according to shared bastion hosts, target assets, administrator privileges, session-control weaknesses, and policy violations. After one remediation batch, high-risk database administration paths decreased from 486 to 173, with a reduction of 64.4%. Incomplete session-audit cases also decreased by 2,140 records. Compared with severity-only ranking, the proposed method avoided early remediation on vulnerable access components that were not connected to sensitive assets. Compared with asset-only ranking, it avoided over-prioritizing sensitive databases that had strong session controls and complete command logs. Compared with policy-violation counting, it captured cases where access approval existed but the bastion host or audit path remained weak. These findings indicate that remediation should be organized around privileged-access chains rather than isolated vulnerability records.

3.3 Scoring Efficiency and Score Stability

The median risk-scoring time was 33 ms per session, and the full privileged-access assessment was completed in 9.7 min. This runtime is suitable for large cloud operation environments with frequent administrator sessions and changing access policies. The model also showed stable scoring across four access-policy snapshots. Risk-score drift remained below 0.16 points on a ten-point scale, indicating that the model was not overly sensitive to minor policy changes. This stability is important in cloud operations because administrator roles, emergency approvals, bastion-host mappings, and database access rules often change [28,29]. A risk model that changes sharply after small policy updates may produce unstable remediation priorities. In contrast, the proposed model kept the main risk order while still reflecting changes in session duration, command-audit status, just-in-time compliance, and target sensitivity. Compared with general zero-trust access control models, which mainly focus on access decisions and policy enforcement, this method provides a direct risk score for each privileged session. Therefore, it can support both access governance and vulnerability remediation planning.

3.4 Comparison with Existing Studies and Security Implications

The results show that policy-driven risk quantification provides a different view from conventional vulnerability management, privileged user auditing, and zero-trust access control. Existing privileged-access studies have improved account protection, session recording, behavior classification, and audit evidence. Zero-trust studies have strengthened least privilege, continuous verification, and policy-based authorization. These methods are useful, but they do not fully explain which vulnerable privileged sessions should be fixed first when thousands of administrator sessions, bastion hosts, production assets, and databases are involved. The present study addresses this gap by treating the privileged-access chain as the main assessment unit. The findings indicate that a vulnerable bastion host becomes more critical when it is used for sensitive database maintenance, when the administrator has cross-environment privileges, or when command logs

are incomplete. Similarly, a long emergency session may not be the highest-risk case if it is fully approved, limited in scope, and completely audited [30]. These results support a policy-driven scoring method that evaluates vulnerability status, privilege scope, asset sensitivity, session control, and audit completeness together. The remediation effect and scoring logic are summarized in Fig. 2.

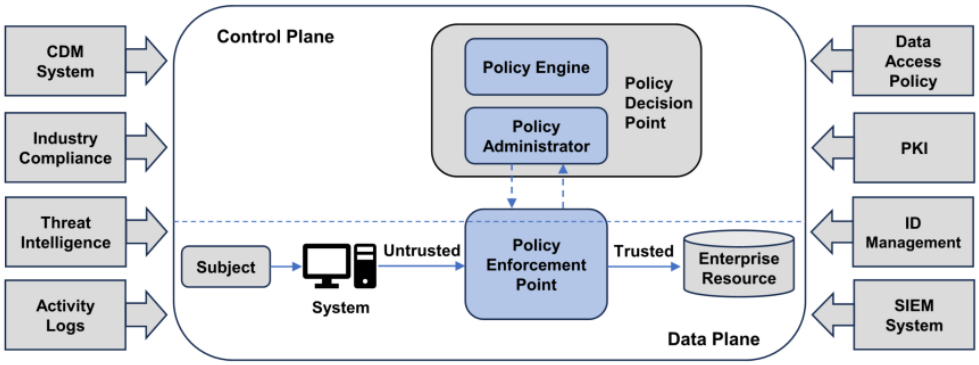

Fig. 2. Policy-driven risk scoring of privileged access using identity attributes, session context, policy status, and audit completeness.

4. Conclusion

This study proposed a policy-driven risk quantification framework for vulnerable privileged access sessions in cloud infrastructure. The framework measured session risk by combining privilege level, target asset sensitivity, bastion-host vulnerability status, command-audit completeness, session duration, just-in-time access compliance, and policy violation records. The results show that privileged-access risk cannot be explained by vulnerability severity alone. The proposed method identified 1,360 high-risk privileged-access chains from 76,000 session records and 13,600 vulnerability findings. It further grouped these findings into 1,890 remediation cases. After one remediation batch, high-risk database administration paths decreased from 486 to 173, and incomplete session-audit cases decreased by 2,140 records. The median scoring time was 33 ms per session, and the full assessment was completed in 9.7 min. These results indicate that the method can be used in large cloud operation environments. The main contribution of this study is that it uses the privileged-access chain as the main unit of risk assessment, instead of assessing vulnerabilities, assets, policies, and audit records separately. This method can help identify vulnerable access paths that may affect cloud data confidentiality and operational accountability. It can also support bastion-host hardening, privileged access review, database access governance, and remediation planning. However, this study has some limitations. The experiments were based on structured cloud operation records, and the scoring weights were adjusted through sensitivity analysis rather than long-term production feedback. Future studies should test the framework in live cloud environments, include more real attack and misuse evidence, and refine the scoring weights using actual remediation results.

References

1. Qi, C., & Qiao, X. (2026). Efficient Data Sampling and Feature Selection Algorithms for Scalable Machine Learning Pipelines.

2. Ghosh, S., Shetty, M., Bansal, C., & Nath, S. (2022, November). How to fight production incidents? an empirical study on a large-scale cloud service. In Proceedings of the 13th Symposium on Cloud Computing (pp. 126-141).

3. Shao, W. (2026). Policy-Driven Vulnerability Risk Quantification framework for Large-Scale Cloud Infrastructure Data Security. arXiv preprint arXiv:2604.06252.

4. Su, D., & Shi, X. Optimizing Family–School Collaboration to Improve Educational and Social Outcomes for Children with Intellectual Disabilities in Inclusive Public Schools.

5. Ahmadi, S. (2024). Zero trust architecture in cloud networks: Application, challenges and future opportunities. Ahmadi, S.(2024). Zero Trust Architecture in Cloud Networks: Application, Challenges and Future Opportunities. Journal of Engineering Research and Reports, 26(2), 215-228.

6. Ma, Y. (2026). Industrial Financial Risk Prediction Model Based on Graph Neural Network and Knowledge Graph Inference. Journal of Circuits, Systems and Computers, 35(16), 2650103.

7. Gao, G., Gao, R., Gao, R., Zhou, H., & Lu, C. (2026, March). Performance Study of Enterprise SMS Communication Scheduling Mechanisms in Triple-Play Convergence Environments. In 2026 IEEE 8th International Conference on Communications, Information System and Computer Engineering (CISCE) (pp. 82-86). IEEE.

8. Chaudhari, S. (2023). Mitigating Insider Threats in SaaS PEO Applications through Behaviour-Based Access Control. Available at SSRN 5277381.

9. Li, Y., & Liu, S. (2026, May). A Study on Dynamic Optimization of Alerting Policies and Multi-Agent Decision-Making Mechanisms in Cloud Environments. In 2026 7th International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT) (pp. 703-706). IEEE.

10. Hakala, H. (2026). Requirements elicitation for a privileged access management system in a large manufacturing company (Doctoral dissertation, University of Jyväskylä).

11. Zhao, J., Fan, J., & Li, L. (2026). A Study on an Explainable Causal-Enhanced LLM Agent for Predicting the Forming Quality of Automotive Component Materials.

12. Fu, Y., Gui, H., Li, W., & Wang, Z. (2020, August). Virtual Material Modeling and Vibration Reduction Design of Electron Beam Imaging System. In 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA) (pp. 1063-1070). IEEE.

13. Jacobs, J., Romanosky, S., Suciu, O., Edwards, B., & Sarabi, A. (2023, July). Enhancing vulnerability prioritization: Data-driven exploit predictions with community-driven insights. In 2023 IEEE European symposium on security and privacy workshops (euroS&pW) (pp. 194-206). IEEE.

14. Chen, H., Ning, P., Li, J., & Mao, Y. (2025). Energy Consumption Analysis and Optimization of Speech Algorithms for Intelligent Terminals.
15. Liang, R., Fan, F., Liang, Y., & Li, S. (2025). Constructing an Adaptive Optimization Model for Ribbon Recommendation and Interface for User Habits.
16. Ahmed, I., & Sohrab, T. B. (2023). AI-Driven Vulnerability Prioritization for Enterprise Networks: A Quantitative Study Using Attack-Graph Models. American Journal of Advanced Technology and Engineering Solutions, 3(04), 129-166.
17. Liu, H., Xu, D., Ma, Q., Xu, S., & Qiu, D. (2026). Memory Poisoning Propagation and Repair Mechanism in Multi-Agent Collaborative Environments.
18. Syed, S. (2025). Privilege Access Management as a Cornerstone of National Security: Protecting Critical Infrastructure and Government Systems. Journal Of Multidisciplinary, 5(7), 899-906.
19. Yin, J., Huang, Y., & Rao, H. (2026). A Study on the Dynamic Evolution of Learning Behavior and Outcome Prediction in Digital Educational Environments. Available at SSRN 6607139.
20. Yang, J. (2026). Stage-Coupled Computational Framework for Stratified Accessibility and Equity Analysis in Community-Based Elderly Care Services.
21. Sitharaman, S., Karim, H., Gupta, D., & Tyagi, M. (2025). Scalable Privilege Analysis for Multi-Cloud Big Data Platforms: A Hypergraph Approach. arXiv preprint arXiv:2511.15837.
22. Deng, X., & Yang, J. (2026). Design of a Quantitative Futures Trading Model Incorporating Multimodal Feature Fusion and Tail Risk Control. Available at SSRN 6702398.
23. Haque, M. A., Gupta, K. D., & Zonouz, S. Comparative Cybersecurity Vulnerability Assessment of US Critical Infrastructure: A Composite Index Approach. Available at SSRN 6676872.
24. Zhang, Z. (2026). A Study on Return Optimization in E-commerce for Complex Consumer Goods Driven by Installation Information Quality. Available at SSRN 6734720.
25. Behringer, F., & Baumann, P. (2025). Integrating identity and access management and privileged access management for enhanced identity security in financial institutions: A zero-trust approach. Cyber Security: A Peer-Reviewed Journal, 9(2), 114-129.
26. Zhang, Z., Gao, Y., & Tong, Y. (2026). CausalML4CX: A Causal Inference and Machine Learning Framework for Customer Experience Optimization with Application in Banking.
27. Seyedkazemi Ardebili, M., & Bartolini, A. (2026). KubeIntellect: A Modular LLM-Orchestrated Agent Framework for End-to-End Kubernetes Management: MS Ardebili, A. Bartolini. Journal of Grid Computing, 24(3), 17.
28. Xu, T., Zhang, J., & Zhu, W. (2026). Reproducible Modeling Pipelines and Cross-Window Stability in Subprime Auto Loan Credit Risk Assessment. Available at SSRN 6893861.
29. Fang, X., Yang, Y. H., & Wang, S. (2026). Energy-Efficient Context-Aware Multimodal AI Inference at the Edge.

30. Pramono, A., Hikmawati, A., Hartiningtiyaswati, S., & Smith, J. (2025). Breastfeeding support and protection during natural disaster and climate-related emergencies in Indonesia: policy audit. Journal of Human Lactation, 41(2), 231-242.