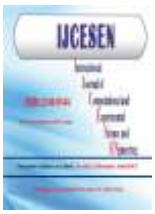
International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 12-No.1 (2026) pp. 397-404
http://www.ijcesen.com



ISSN: 2149-9144

Research Article

A Compliance-Driven Framework for Data Curation and Gating in Machine Learning Training for Enterprise Privacy Infrastructure

Projjal Ghosh*

Independent Researcher, USA
* **Corresponding Author Email:** projjalkg@gmail.com - **ORCID:** 0000-0002-5247-0850

Article Info:

DOI: 10.22399/ijcesen.4805
Received : 28 November 2025
Revised : 10 January 2026
Accepted : 13 January 2026

Keywords

Federated Learning,
Privacy-Preserving Machine Learning,
Data Governance,
Usage Control,
Anomaly Detection,
Blockchain Provenance

Abstract:

Large-scale machine learning systems spread over distributed infrastructures are confronted with crucial issues of managing sensitive data and, at the same time, abiding by regulatory requirements. In general, training pipelines do not have the means by which they can monitor the way in which protected data is introduced to model development; thus, there are quite significant privacy risks in decentralized environments. Also, the lack of complete visibility hinders the organizations' capability to trace data sources, grasp the movement of information between the systems, and check the conformity to the compliance requirements. In many cases, sensitive data is not properly safeguarded and is even allowed to be exploited beyond authorized purposes, both during training and inference stages. Automated classification systems detect sensitivity indicators within datasets and apply metadata tags specifying permissible uses at the precise moment information feeds into training operations. Gating mechanisms function as policy enforcement points that validate access requests against predefined rules, ensuring models access only data appropriate for declared purposes. Attribute-based access control looks at a variety of factors that include attributes of the subject, classes of the resources as well as certain conditions of the environment, and, based on all these factors, it dynamically makes the decision about the authorization. Machine learning anomaly detection is a kind of vigilant system that constantly watches the access patterns and, through behavioral analysis, it can pinpoint the variations from already established compliance standards. Distributed logging that is supported by blockchain keeps very detailed and at the same time very secure audit trails that enable, in the future, the checking of data usage throughout the lifecycle of the models.

1. Introduction

Enterprise use of AI in distributed systems has led to a pressing demand for privacy safeguards that are not only reliable but also able to handle sensitive data in a decentralized manner across different nodes. The organizations that are in charge of the personally identifiable information are under very tight and strict regulatory requirements; however, the traditional training pipelines hardly offer adequate ways through which one can control the manner of data entering and affecting the development of the models. The fundamental challenge extends beyond simple access control to encompass comprehensive visibility into machine learning operations across platforms, teams, and data sources.The core problem manifests in multiple dimensions. Organizations struggle to map where models undergo training, which platforms host these processes, and what data sources feed into them. Upstream data originates from diverse sources, including real-time request streams, outputs from preceding models, and processed datasets extracted from enterprise warehouses. Without the understanding of provenance and data flow between various systems, compliance verification of ingested data is practically impossible. Traditional federated learning architectures are designed to enhance data privacy as all the training data is to be kept on client devices, while only aggregated model updates are shared with central servers. However, gradient updates transmitted during federated optimization can leak sensitive information about individual training samples through various attack vectors. Truex et al.

demonstrate that hybrid approaches combining secure multi-party computation with federated learning provide stronger privacy guarantees by encrypting model updates before aggregation [1]. The proposed hybrid framework prevents the central server from observing individual client contributions in plaintext form. Secure aggregation protocols ensure that only the combined gradient update becomes visible to the coordinating entity. This architectural modification addresses critical vulnerabilities in standard federated learning where honest-but-curious servers could reconstruct training data from repeated gradient observations. Furthermore, the absence of sensitivity metadata at the point of training ingestion means enterprises cannot distinguish between general-purpose data and information requiring restricted usage.

This gap creates substantial risks in distributed environments where data movement across nodes amplifies exposure potential. Even when sensitivity classifications exist, ensuring a model's declared purpose remains consistent from training through deployment and inference presents additional complexity. A fraud detection model trained on personally identifiable information must maintain its security-focused intent throughout its lifecycle, preventing repurposing for marketing optimization or other non-compliant applications. Differential privacy mechanisms offer mathematical guarantees for privacy preservation in federated settings through controlled noise injection. Geyer et al. examine client-level differential privacy where noise addition occurs at individual participants before model update transmission [2]. This approach contrasts with server-side differential privacy that aggregates updates before applying noise. Client-level protection provides stronger privacy guarantees since the server never observes unprotected gradients from any participant. The privacy budget allocation becomes critical in determining the trade-off between model accuracy and individual privacy protection. Careful calibration ensures sufficient noise prevents membership inference attacks while maintaining acceptable model utility. The challenge intensifies in multi-tenant cloud environments where model serving infrastructure processes requests from diverse applications, making runtime policy enforcement critical for preventing unauthorized data access during inference operations.

This paper addresses these challenges by proposing a compliance-driven framework for data curation and gating in machine learning training environments. The framework implements policy enforcement points that identify, classify, and restrict data usage, ensuring sensitive elements serve only designated purposes aligned with platform safety and security objectives. The contributions include a modular architecture for data curation incorporating real-time flow mapping, gating protocols integrated with anomaly detection for continuous compliance monitoring from training through inference, and evaluation methodologies for privacy preservation in distributed system contexts.

2. Related Work

On the one hand, federated learning architectures allow for joint model training over distributed nodes while at the same time local data remains on each node, thereby lessening privacy risks associated with centralization. But on the other hand, typical federated learning models are not free of gradient leakage issues, that is to say that the adversaries may reconstruct training examples from shared model updates. Hybrid frameworks combining secure multi-party computation with federated optimization provide enhanced protection by encrypting gradient information before aggregation at coordination servers. Communication efficiency emerges as a critical constraint in distributed settings where bandwidth limitations and intermittent connectivity impact convergence behavior. Federated averaging algorithms reduce synchronization overhead by performing multiple local training iterations before parameter exchange, proving effective even under non-identical data distributions across participants.

Security threats from Byzantine participants necessitate robust aggregation mechanisms capable of identifying and excluding malicious contributions. Geometric approaches to computing distances between submitted gradients enable outlier detection, ensuring corrupted updates are excluded from global model formation. Usage control frameworks extend traditional access control by enforcing ongoing obligations and conditions throughout data utilization periods rather than terminating enforcement after initial access grants. Mutable attributes change dynamically based on subject actions, affecting continuous authorization decisions during extended access sessions.

Anomaly detection techniques identify deviations from established behavioral patterns through unsupervised learning approaches that require no labeled violation examples. Extending isolation techniques through the use of hyperplanes at arbitrary angles enhances the overall capability of detection regardless of features being oriented in different directions. By using blockchain-based provenance systems, any modifications of previously recorded transactions are obviated

through the creation of unalterable audit trails via cryptographic hash chains and distributed consensus, while smart contracts facilitate the granularity and ease with which access rights are managed.

3. Framework Architecture

3.1 Distributed System Design

The framework operates on a federated architecture where data remains distributed across edge nodes while training occurs collaboratively without full centralization. Such an arrangement complies with data residency regulations while at the same time it does not impede the progression of model development across different organizations. The central challenge involves minimizing communication costs while achieving convergence comparable to centralized training approaches. McMahan et al. address this challenge through the Federated Averaging algorithm, which reduces communication rounds by allowing clients to perform multiple local gradient descent updates before synchronization [3]. Each participating client downloads the current global model from a central server and executes several training iterations on local data. After completing local training epochs, clients transmit only the updated model parameters back to the server rather than raw training data. The server aggregates these parameter updates through weighted averaging based on the number of local training examples at each client. This approach proves particularly effective when local datasets exhibit non-identical and independent distributions across clients. Statistical heterogeneity represents a fundamental characteristic of federated settings where different organizations or devices naturally possess distinct data distributions. The algorithm maintains convergence properties even under these challenging conditions by balancing local adaptation with global consistency. A visibility layer overlays the core infrastructure, mapping relationships between platforms, operational teams, data sources, and information flows to provide comprehensive oversight of machine learning operations.

The three components of the architecture are interconnected and function together. The data curation pipeline is the part of the system that is responsible for the intake and classification of data, the gating mechanism implements the policy-based access control, and the compliance auditing layer records all data interactions in a way that is resistant to tampering. Distributed training systems face security threats from Byzantine participants who may inject poisoned updates to compromise model integrity. Standard averaging-based aggregation methods remain vulnerable to such attacks since malicious updates receive equal weight with honest contributions. Blanchard et al. propose Krum, a Byzantine-tolerant aggregation rule that identifies and excludes suspicious gradients before model updates [4]. The Krum algorithm operates by computing pairwise Euclidean distances between all submitted gradient vectors from participating clients. For each gradient, the algorithm sums distances to its nearest neighbors to produce a score reflecting geometric centrality within the gradient distribution. The gradient with the minimum score becomes selected as the aggregated update since it lies closest to the majority cluster of honest contributions. This geometric median approach provides robustness against adversarial manipulation even when a significant fraction of participants behaves maliciously. The selection mechanism ensures that outlier gradients submitted by Byzantine clients are excluded from the aggregation process. These components communicate through secure channels, with the visibility layer aggregating metadata to enable holistic compliance verification across distributed nodes.

3.2 Gating Mechanism and Policy Enforcement Rule-Based Access Control

Gates function as policy enforcement points positioned before data enters training phases and during inference operations. A rule-based policy engine validates each data access request against predefined compliance rules, cross-referencing the requesting model's declared purpose with permissible uses specified in data metadata. When a model trained for fraud detection requests customer demographic data tagged for security purposes only, the gate permits access. Conversely, requests from marketing optimization models for the same data result in automatic denial. Attribute-based access control represents a logical evolution beyond traditional role-based models by incorporating contextual attributes into authorization decisions. Hu et al. define attribute-based access control as a paradigm where subject requests to perform operations on objects are granted or denied based on assigned attributes of the subject, object, environment conditions, and policies [5]. Subject attributes characterize requesting entities through properties such as organizational affiliation, security clearance, and job function. Object attributes describe protected resources including data classification level, ownership, and handling restrictions. Environment attributes capture operational context such as current system threat

level, time of day, and network location. The policy engine evaluates Boolean combinations of these attributes to render access decisions dynamically. This approach enables expression of complex authorization requirements that static role assignments cannot accommodate. Policies can specify that analysts may access customer data only during business hours when accessing from internal networks and only for declared fraud investigation purposes. The flexibility supports fine-grained access control necessary for compliance-driven environments where data usage restrictions depend on multiple contextual factors.

The gating system maintains intent fidelity throughout the model lifecycle. A model's declared purpose at training time becomes an immutable attribute verified during deployment and inference. This prevents purpose drift, where models initially trained for legitimate security applications undergo repurposing for non-compliant use cases. Runtime checks during inference validate that input data and output destinations align with the model's original compliance designation. Cryptographic mechanisms ensure the integrity of model metadata throughout deployment pipelines. Digital signatures bind declared purposes to model artifacts, making unauthorized modifications detectable through signature verification. Hash chains establish auditable lineage connecting successive model versions from initial training through production deployment.

3.3 Anomaly Detection Integration

Machine-learning-powered anomaly detection keeps an eye on data access patterns without delay and immediately raises compliance issues that differ from the established norms. The detection system analyses the frequency of requests, data volume, temporal patterns, and profiles of the requesting entities to indicate behavior that might be abnormal. For instance, sudden requests for large volumes of location data from a model previously accessing only aggregated statistics triggers investigation workflows. Isolation-based anomaly detection provides efficient identification of outliers in high-dimensional spaces without requiring labeled training data. Hariri et al. introduce Extended Isolation Forest, which improves upon standard isolation methods by using hyperplanes with random slopes rather than axis-parallel splits [6]. Traditional isolation forests partition feature spaces through splits parallel to coordinate axes, creating bias toward certain types of anomalies while missing others. The extended approach generates random hyperplanes at arbitrary angles, enabling more effective isolation of anomalies regardless of their orientation in feature space. Each tree in the forest selects a random normal vector and an intercept to define splitting hyperplanes. Branch extensions partition remaining samples recursively until isolation occurs or maximum depth is reached. Anomaly scores derive from average path lengths across all trees in the forest. Observations requiring fewer splits for isolation receive higher anomaly scores since outliers naturally separate from dense regions with minimal partitioning. The algorithm exhibits linear time complexity with respect to dataset size, making it suitable for streaming access pattern analysis in distributed systems.

This monitoring extends beyond simple threshold detection to incorporate behavioral analysis, recognizing subtle compliance violations. The system learns normal access patterns for each model class and deployment context, enabling detection of sophisticated attempts to circumvent gating policies through gradual boundary expansion or indirect data acquisition. Temporal analysis identifies drift in access behavior unfolding incrementally over extended periods. Sudden volume spikes may indicate obvious violations, but gradual increases designed to avoid detection thresholds require more sophisticated monitoring approaches. Contextual evaluation considers appropriateness of access patterns relative to operational norms for specific model types and organizational contexts.

4. Compliance Auditing and Verification

4.1 Distributed Logging Infrastructure

A secure distributed logging system records comprehensive metadata for all data access and usage events across the federated infrastructure. Each log entry captures requesting entity identity, accessed data classification, timestamp, declared purpose, and gating decision rationale. The logging system employs tamper-evident data structures ensuring audit trail integrity, with cryptographic proofs enabling verification of record authenticity during regulatory reviews. Blockchain technology offers decentralized infrastructure for maintaining immutable provenance records in cloud computing environments. Liang et al. propose ProvChain, a blockchain-based architecture that addresses privacy and availability challenges in cloud data provenance systems [7]. Traditional provenance tracking in cloud environments faces vulnerabilities from centralized storage where malicious administrators or compromised nodes could alter historical records. The ProvChain architecture distributes provenance data across blockchain

nodes, eliminating single points of control that enable tampering. Each provenance record undergoes hashing before insertion into blockchain transactions, creating cryptographic fingerprints that detect any subsequent modifications. The system employs a two-layer storage structure separating large provenance documents from blockchain metadata to address scalability limitations. Detailed provenance information resides in distributed cloud storage while the blockchain maintains only cryptographic hashes and access control metadata. This hybrid strategy serves the purpose of a perfect balance between the bank of the immutable records and the storage efficiency of a voluminous audit log. Smart contracts carry data-sharing policies encoded directly on the blockchain, thus, access restrictions are enforced automatically without the need for trusted intermediaries. The consensus mechanism guarantees that provenance records, thus, get approval only when several blockchain members confirm the transaction's validity. Encryption protects sensitive provenance information stored off-chain while maintaining verifiability through hash comparisons. The system architecture allows for very detailed access control such that data owners can even indicate which attributes of provenance records external parties may query. The cooperation with corporate identity management systems makes it possible to have role-based access control even for audit data, and thus compliance officers can be granted access to the relevant records, which at the same time keep the most sensitive audit information away from unauthorized persons. Privacy measures such as differential privacy facilitate the querying of audit logs in such a way that individual data access patterns are not exposed thus giving a balance between transparency requirements and privacy preservation. Selective disclosure mechanisms allow compliance verification without revealing the complete audit trail that can be used to identify the sensitive details of the operation.Zero-knowledge proofs allow demonstration of compliance properties without disclosing underlying transaction data.

4.2 Retrospective Analysis Capabilities

The auditing layer supports retrospective analysis verifying whether historical data usage upheld platform safety and security objectives. Compliance officers can query the audit trail to reconstruct complete data lineage for specific models, confirming that sensitive information served only designated purposes throughout training and deployment. Cross-system flow validation enables tracking of data transformations across processing stages, ensuring derived datasets maintain appropriate sensitivity classifications. Provenance systems capture the history of data artifacts through their creation, transformation, and usage across computational workflows. Pérez et al. conduct a systematic review of provenance approaches across scientific computing, database systems, and workflow management platforms [8]. Provenance information serves multiple purposes, including reproducibility verification, quality assessment, and compliance auditing. Retrospective provenance traces backward from outputs to identify all inputs and transformations contributing to final results. Prospective provenance describes intended computational processes before execution, enabling validation that actual workflows conform to designed procedures. The Open Provenance Model establishes standard representations for provenance graphs through entities, activities, and relationships. Entities represent data artifacts at various processing stages, while activities describe computational processes that consume and produce entities. Derivation relationships link the outputs with their source inputs through intermediary processing steps. Provenance granularity is the level of detail that is captured, starting from very general workflow-level tracking and going to very detailed recording of individual data element transformations. The means of storage have to make trade-offs between the full capture and the system overhead. Inline provenance collection intercepts data operations in real-time, ensuring complete capture at the cost of runtime performance impact. Offline provenance reconstruction analyzes logs and metadata after execution completes, reducing overhead but potentially missing information not preserved in available records. Query capabilities enable investigation of lineage questions, such as identifying all datasets derived from specific sources or determining which processes produced particular outputs.

Intent enforcement verification confirms that models maintained their declared purposes across lifecycle phases. Automated compliance reports pull together the audit data to show adherence to regulations, and at the same time, they point out any gating violations or anomalous access patterns that require further investigation. This look-back feature is a must-have during regulatory audits when organizations need to show that they have been continuously compliant rather than only at certain points in time. Visualization software converts provenance graphs into interactive charts that present complex dependency relationships, and thus, the analysts can easily move through them.

Temporal analysis examines how data usage patterns evolve across extended periods, detecting gradual deviations from authorized compliance boundaries. Comparative analysis identifies anomalous lineage patterns that differ from established norms for similar model types or organizational contexts.

Table 1. *Federated Learning Communication Strategies and Privacy Mechanisms [3], [4]*

Aspect	Federated Averaging	Byzantine-Tolerant Aggregation
Communication Pattern	Multiple local training iterations before synchronization	Single gradient submission per round
Aggregation Method	Weighted averaging based on local dataset sizes	Geometric median selection from gradient clusters
Convergence Behavior	Effective under non-identical data distributions	Robust against statistical heterogeneity
Privacy Protection	Local data remains on client devices	Prevents poisoning through outlier exclusion
Threat Model	Honest-but-curious central server	Adversarial Byzantine participants
Computational Overhead	Reduced through local iteration batching	Increased due to pairwise distance computation
Network Efficiency	Minimizes communication rounds significantly	Standard synchronization frequency
Security Guarantee	Gradient-level information leakage is possible	Malicious update detection and filtering

Table 2. *Access Control Mechanisms and Policy Enforcement Frameworks [5].*

Component	Attribute-Based Access Control	Usage Control Enforcement
Authorization Basis	Subject, object, and environmental attributes	Continuous obligations and conditions
Policy Evaluation	Boolean combinations of attribute predicates	Temporal constraints throughout access sessions
Enforcement Timing	Decision point before access grant	Ongoing monitoring during data utilization
Attribute Mutability	Static attributes at authorization time	Dynamic attributes changing with system state
Contextual Factors	Security clearance, organizational role, network location	Time windows, geographic restrictions, privacy budgets
Obligation Types	Pre-authorization requirements only	Pre, ongoing, and post-access mandates
Policy Expression	Declarative attribute rules	Temporal logic with state transitions
Compliance Verification	Point-in-time access decision validation	Continuous adherence checking across the lifecycle

Table 3. *Anomaly Detection Techniques for Access Pattern Monitoring [6].*

Characteristic	Extended Isolation Forest	Network Behavioral Analysis
Learning Paradigm	Unsupervised outlier detection	Semi-supervised pattern recognition
Training Requirements	No labeled anomaly examples needed	Limited labeled data with abundant normal samples
Detection Mechanism	Hyperplane-based feature space partitioning	Sequential pattern modeling and deviation measurement
Computational Complexity	Linear time with dataset size	Dependent on the network architecture depth
Feature Space Handling	Arbitrary orientation hyperplanes	Hierarchical feature extraction
Temporal Dependencies	Independent observation analysis	Recurrent structures for sequence modeling
Anomaly Scoring	Average path length across forest trees	Reconstruction error or prediction deviation
Adaptability	Effective across diverse feature orientations	Learns domain-specific normal patterns

Table 4. *Provenance and Audit Trail Infrastructure Components [7, 8].*

Element	Blockchain-Based Provenance	Traditional Provenance Systems
Storage Architecture	Two-layer with off-chain detailed records	Centralized or distributed databases
Immutability Guarantee	Cryptographic hash chains with consensus	Administrative access controls
Tamper Detection	Automatic through hash verification	Periodic integrity audits
Access Control	Smart contract enforcement	Role-based permission systems
Granularity Options	Fine-grained attribute-level tracking	Workflow-level or entity-level capture
Query Capabilities	Backward and forward lineage reconstruction	Retrospective and prospective provenance
Privacy Preservation	Selective disclosure with encryption	Differential privacy on aggregate queries
Scalability Approach	Hybrid storage separating metadata from content	Compression and optimization strategies

5. Conclusions

The compliance-driven framework presented addresses fundamental challenges in managing sensitive data across distributed machine learning infrastructures. Traditional approaches to data governance prove insufficient when training occurs across multiple nodes with heterogeneous data sources and varying security postures. The integration of data curation pipelines with gating mechanisms provides organizations with fine-grained control over how protected information influences model development. Automated classification during ingestion ensures that sensitivity metadata remains current and accurately reflects regulatory requirements at the moment data enters training systems. Policy enforcement points positioned before training and during inference operations prevent unauthorized access by validating declared purposes against permissible uses encoded in resource metadata. Attribute-based access control extends beyond static role assignments to incorporate dynamic evaluation of contextual factors, supporting complex compliance requirements that evolve with operational conditions. Machine learning anomaly detection complements rule-based enforcement by identifying subtle violations that manifest through gradual boundary expansion or indirect acquisition patterns. The behavioral analysis capabilities enable the detection of sophisticated circumvention attempts that simple threshold monitoring would miss. Blockchain-based logging infrastructure ensures audit trail integrity through cryptographic hash chains and distributed consensus mechanisms that make retroactive tampering computationally infeasible. The hybrid storage architecture balances immutability guarantees with practical scalability by maintaining detailed provenance information off-chain while recording verification hashes on the blockchain. Provenance tracking capabilities support retrospective analysis essential for regulatory audits, enabling reconstruction of complete data lineage from training through deployment phases. Organizations can demonstrate continuous compliance rather than point-in-time assessments by querying audit trails to verify that historical data usage aligned with declared purposes. The framework addresses scalability considerations through communication-efficient federated learning algorithms that reduce synchronization overhead while maintaining convergence properties under statistical heterogeneity. Byzantine-tolerant aggregation provides resilience against adversarial participants who might attempt to poison models through malicious update injection. The geometric median approach excludes outlier contributions that deviate significantly from honest participant clusters. Future developments should focus on automated policy generation through machine learning techniques that extract compliance rules from regulatory text and historical audit patterns. Empirical validation in production enterprise environments would provide valuable insights into real-world performance characteristics and operational challenges beyond conceptual frameworks. Ethical considerations require careful examination to ensure that gating algorithms do not introduce bias or create unfair barriers to legitimate model development activities. The framework establishes practical pathways toward trustworthy artificial intelligence systems that balance innovation imperatives with regulatory compliance demands across privacy-sensitive domains.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial

interests or personal relationships that could have appeared to influence the work reported in this paper

- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.

- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1] Stacey Truex et al., "A Hybrid Approach to Privacy-Preserving Federated Learning," arXiv, 2019. [Online]. Available: https://arxiv.org/pdf/1812.03224

[2] Robin C. Geyer et al., "Differentially Private Federated Learning: A Client Level Perspective," arXiv, 2018. [Online]. Available: https://arxiv.org/pdf/1712.07557

[3] H. Brendan McMahan et al., "Communication-Efficient Learning of Deep Networks from Decentralized Data," Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, 2017. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a/mcmahan17a.pdf

[4] Peva Blanchard et al., "Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent," 31st Conference on Neural Information Processing Systems, 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/f4b9ec30ad9f68f89b29639786cb62ef-Paper.pdf

[5] Vincent C. Hu et al., "Attribute-Based Access Control," IEEE COMPUTER SOCIETY, 2015. [Online]. Available: https://profsandhu.com/cs5323_s18/Hu-2015.pdf

[6] Sahand Harir et al., "Extended Isolation Forest," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, 2021. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8888179

[7] Xueping Liang et al., "ProvChain: A Blockchain-based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability," 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2017. [Online]. Available: https://www.researchgate.net/profile/Sachin-Shetty/publication/317182541

[8] Beatriz Pérez et al., "A systematic review of provenance systems," Springer Nature, 2018. [Online]. Available: https://www.researchgate.net/profile/Carlos-Saenz-Adan/publication/323242431_A_systematic_review_of_provenance_systems/links/5b34ae1caca2720785effb1a/A-systematic-review-of-provenance-systems.pdf