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

Vol. 11-No.4 (2025) pp. 8234-8240
http://www.ijcesen.com

ISSN: 2149-9144

Research Article

Analytics Data Sanitization Framework for Event Tracking Systems

Laxmi Deepthi Atreyapurapu*

Independent Researcher, USA.
* **Corresponding Author Email:** deepthiatreya@gmail.com.- **ORCID:** 0000-0002-5247-0050

Article Info:

DOI: 10.22399/ijcesen.4208
Received : 03 September 2025
Accepted : 22 October 2025

Keywords

Analytics Data Sanitization,
Privacy-Enhancing Technologies,
Event Tracking Systems,
Data De-Identification,
Centralized Processing Architecture,
Regulatory Compliance

Abstract:

Abstract should be about 100-250 words. It should be written times new roman and 10 punto. Contemporary digital ecosystems produce enormous behavioral data sets with event tracking systems that guide strategic choices and refine user experience. Nevertheless, the process of gathering analytics data creates inherent conflicts between obtaining useful insights and safeguarding sensitive personal data from improper disclosure. Analytics systems that are fed raw event data pose significant privacy threats by having personally identifiable data, financial credentials, authentication tokens, or health records unwittingly sent to third-party services. The repercussions of poor data management go beyond technical breakdowns to include regulatory sanctions, legal liabilities, and loss of user trust that effectively compromise organizational longevity. This article offers a detailed framework for the application of analytics data sanitization as a core architectural element, as opposed to an afterthought in system development. The architecture includes systematic enumeration of sensitive data elements in event payloads, creation of transformation rules balancing privacy protection and analytical utility, and deployment of centralized processing frameworks that support consistent enforcement in all analytics integrations. Cryptographic hashing, partial masking, and complete replacement transformation techniques cover various situations where data has legitimate analytical purposes or is pure risk with no value. Centralized sanitization layers provide single points of control where transformation logic is kept consistent, avoiding erratic implementations across distributed system elements. Organizations that follow robust sanitization frameworks gain regulatory compliance, limit data breach exposure, maintain analytical capabilities, and uphold consumer trust by being capable of demonstrating privacy commitments.

1. Introduction

Current virtual products rely closely on behavioral analytics for making strategic decisions, improving user experience, and streamlining product development. Event tracking systems are the building blocks of understanding user behavior through identifying actionable insights in intricate behavioral patterns from millions of user sessions every day. The digital economy has experienced accelerated growth in data accumulation, as contemporary applications gather enormous amounts of event data to enable decision-making activities across various fields such as entertainment networks, financial services, healthcare networks, and enterprise software. This expansion of data gathering practices has also brought about significant privacy threats through the unintentional transfer of personal information to third-party analytics platforms.The vulnerability of apparently anonymized datasets has been shown through extensive research testing the resilience of de-identification methods. A seminal study profiling a dataset comprising subscriber records found that individual users could be successfully re-identified when anonymized behavior patterns were correlated against publicly accessible auxiliary information. The research showed that knowing about eight movie ratings and their dates, with a window of precision of fourteen days, was enough to identify the subscribers in the anonymized set with high certainty. This de-anonymization was possible despite the set having stripped explicit identifiers and having more than four hundred thousand subscriber records covering around one hundred million ratings. The attack model took advantage of the statistical rarity of personal tastes and temporal patterns and showed that mere

behavioral data are a potent identifier that can breach privacy if combined with auxiliary information [1].The implications of poor handling of data go far beyond technical breakdowns to include legal exposure, regulatory sanctions, and user disillusionment. Historical examination of information security threats revealed that the spread of computer systems and data networks provided unparalleled opportunities for privacy intrusion. Early models of information security identified three broad categories of threats: revelation of information to unauthorized entities, alteration of information by unauthorized actors, and denial of service to rightful users. Ensuring the protection of privacy was realized to necessitated technical protection against both external and insider threats. These fundamental principles of security reaffirmed that privacy protection necessitates end-to-end architectural strategies instead of discrete technical controls [2].The paper suggests a formal framework for applying data sanitization of analytics data as a native architectural feature. Instead of treating privacy as an add-on, the approach places data cleansing as an essential requirement inside the event tracking infrastructure. The framework addresses the proven threats of re-identification by using transformation methods that block correlation attacks while maintaining statistical properties for useful analytics.

2. Sensitive Data Identification Strategy

The cornerstone of successful sanitization lies in thorough identification of sensitive data elements in event payloads. Sensitive data includes authentication credentials such as passwords and security tokens, financial information in the form of payment card information and banking identifiers, government identification numbers utilized for tax or social security purposes, and protected health information under medical privacy laws. The task of locating sensitive data has become more challenging as applications have changed to gather more fine-grained behavioral data, building out large attack surfaces where sensitive data is more likely to leak out unintentionally via apparently harmless event parameters.The landscape of vulnerabilities for sensitive data exposure has been thoroughly described through systematic web application security weakness analysis. The Open Web Application Security Project has a systematically rated catalog of the top ten most important security threats to web applications, where sensitive data exposure is always a basic category of vulnerability. The vulnerability exists through several attack vectors, such as a lack of encryption when data is being transmitted, poor-quality cryptographic mechanisms that can be broken using computational attacks, incorrect key management procedures where encryption keys are kept with the encrypted data, and storage of sensitive data in clear text in databases or log files. Empirical analysis indicates that exposure of sensitive data often occurs due to developers accidentally exposing confidential data in debugging outputs, error messages, or monitoring data that is further sent to external systems without proper sanitization. Statistical analysis of past incidents proves that attackers systematically take advantage of these vulnerabilities, compromised credentials, and financial information, having a high price on dark markets [3].Identification methods integrate human review processes with machine-based detection mechanisms to provide full coverage of intricate application platforms. Security teams thoroughly analyze event schemas, following data flow from points of collection through integrations with analytics. Pattern recognition algorithms are used by automated scanning tools to alert on possible sensitive data by field naming conventions, data type, and content patterns. Structured data types can be recognized using regular expression patterns like social security numbers in nine-digit formats, credit card sequences in standard sixteen-digit patterns, and email addresses with standard internet addressing conventions.Studies of privacy-enhancing tool development have determined thorough frameworks for classifying personal information. The model differentiates between direct identifiers that uniquely identify people with high confidence levels, such as official government-issued identification numbers, biometric data readings, and unique device IDs, and quasi-identifiers that can result in re-identification when aligned with other source information. Such quasi-identifiers include demographic features like age ranges, gender, geographic location at different granularity levels, occupation types, and educational attainment. Contemporary data sets often include rich sets of quasi-identifiers together have discriminatory power as powerful as direct identifiers when attackers have access to auxiliary information on public databases or social networking sites [4].

3. Development of Transformation Rules

After sensitive fields have been identified, the proper transformation rules have to be defined according to analysis needs and privacy limitations. Three major transformation methods respond to various situations faced in practice, each with unique privacy protection versus analytical usefulness trade-offs.

3.1. Cryptographic Hashing

Cryptographic hashing establishes a method for maintaining uniqueness while removing immediate identifiability. For analytics where the need to track unique entities is present without disclosing identities, one-way hash functions convert sensitive identifiers to fixed-length values that remain consistent over repeated observations. Email addresses, user IDs, and device IDs can be hashed to facilitate counting unique users and monitoring patterns of behavior between sessions without storing real personal data.Studies that have tested security vulnerabilities from a data point of view have shown that attacks on training data, parameters, and inference operations by adversaries are essential threats to system integrity. Machine learning systems are especially vulnerable to membership inference attacks, whereby adversaries identify if particular users' data were part of training datasets by observing model outputs to specially designed queries. Cryptographic methods such as homomorphic encryption allow computation over encrypted information without decryption, making it feasible to have privacy-preserving machine learning where sensitive data is never exposed during training and inference. Differential privacy frameworks introduce noise into training data or model predictions to make it difficult for adversaries to determine whether individual contributions came from specific people in datasets. Sanitization of data is an infrastructural security layer that blocks sensitive data from reaching machine learning pipelines, which removes whole classes of attacks that take advantage of identifiable data in training datasets [5].

3.2. Partial Masking

There are some analytic situations that need partial exposure to sensitive data structures while keeping full values safe. Masking methods maintain data format and restricted content while hiding most sensitive data by selective substitution of characters with masking characters. Financial information tends to be helped by this method, where showing terminal digits of payment devices helps customer service situations without revealing full account numbers. Payment card data usually goes through masking that retains the first six digits that identify the issuing institution as well as the last four digits that facilitate customer verification, replacing middle digits with asterisks.

3.3. Full Replacement

Highly sensitive fields that don't have any legitimate analytical use should be entirely removed or replaced with generic placeholders. Authentication credentials, security questions, and other authentication factors fit into this group, where transmission to analytics platforms constitutes unacceptable risk. Replacement guarantees that sensitive fields are recognized in event schemas without revealing actual values, avoiding accidental logging while retaining event structure integrity.Research that has defined benchmarking frameworks for de-identification methods has shown that privacy and utility are inherently conflicting goals. The benchmark framework considers various de-identification methods, such as generalization methods that substitute specific values with more generic categories, suppression methods that eliminate sensitive attributes, and perturbation methods that incorporate controlled alterations to data values while maintaining statistical characteristics. Experiments prove that substituting exact ages with five-year age bins offers strong privacy protection with acceptable utility for demographic studies. Research derives quantitative measures of privacy protection, such as k-anonymity measures, guaranteeing each record to be indistinguishable from at least k minus one other records [6].

4. Centralized Processing Architecture

Successful sanitization must be centrally deployed to allow consistency across every analytics integration. A specialized processing layer captures all outgoing analytics events, transforming them first with rules before data is sent to external platforms. The centralized design overcomes inherent problems in distributed systems when various application components fire analytics events, producing many potential points of failure where sensitive information might unintentionally slip past sanitization controls.The processing layer accepts event payloads as structured data objects, processes all fields using proper transformation rules based on field names and content patterns, and ensures that no unsanitized sensitive data is left behind before sending events to destination platforms. Centralization prevents inconsistent implementation across various analytics integrations, lowers maintenance overhead by routing sanitization logic through reusable components, and creates an explicit audit trail for verification of compliance.Studies investigating formal methods in computer security have found that mathematical modeling allows for accurate specification of security needs and thorough analysis of protection techniques. Access matrix

models depict systems as matrices in which rows are subjects like users or processes, columns are objects like files or data structures, and matrix entries indicate the access rights subjects have for objects. The lattice model of secure information flow establishes security levels as members of a mathematical lattice structure in which partial ordering relations determine which flows of information are allowed on the basis of security classification. Formal models give a rigorous foundation for the design and verification of security mechanisms [7].Implementation generally entails developing utility functions or middleware components that intercept analytics calls across the application codebase. Such components apply sanitization rules transparently without the need for changes to the current event tracking code and offer complete protection via systematic interception of all outbound analytics traffic.Computer privacy problems explored via historical evaluation found underlying tensions between the efficiency benefits of centralized data structures and the privacy risk posed by compiling personal data in machine-readable form. Automated data systems made unprecedented data collection, storage, retrieval, and analysis capabilities available. At the same time, these capabilities posed enormous privacy threats to the availability of personal information for access, copying, transmission, and aggregation across formerly isolated databases. The study found a number of privacy threat categories, such as surveillance threats from systematic observation of individual behavior, aggregation threats from the coordination of information from more than one source, and secondary use threats from organizations reusing data originally collected for one intention to fulfill other purposes. The study acknowledged that centralized control points for access to data allow for more successful implementation of privacy policies than distributed models [8].

5. Organizational Benefits

Adopting structured sanitization frameworks provides significant benefits to various organizational aspects. Compliance with regulatory requirements is enhanced by systematic avoidance of unauthorized transmission of personal data. The framework minimizes organizational risk by eliminating potential vectors of exposure, shielding against data breaches that may be caused by analytics platform vulnerabilities or third-party integration misconfigurations.Operational advantages consist of continued analysis capability in spite of privacy limitations, since properly sanitized data retains behavior patterns and statistical correlations needed for insights creation. Development teams are helped by concise implementation instructions that minimize uncertainty regarding data handling needs. Security teams benefit from visibility into data flows through centralized monitoring points offering end-to-end audit trails. User trust is retained by being able to substantiate dedication to privacy protection, enabling long-term customer relationships and brand integrity.Studies analyzing organizational experience with General Data Protection Regulation compliance have revealed systematic issues that organizations face when trying to meet regulatory demands. Survey statistics indicate that organizations still face serious challenges with core compliance activities. In surveyed organizations, a significant percentage indicated difficulty in having complete inventories of personal data processing activities. The data inventory challenge arises from the decentralized nature of contemporary information systems in which personal data is spread across many databases, applications, file systems, and third-party services. Organizations cited specific difficulties in determining all the places where personal data is stored and what purposes data is processed for. Data subject rights implementation is another major compliance issue, such as the need to allow individuals access to their personal data, accurate corrections, and right to erasure. Technical solutions such as centralized data sanitization frameworks tackle several compliance issues at once by ensuring sensitive data never makes it to external platforms where it generates regulatory risk [9].Data processing migrated to cloud computing platforms raises further considerations for privacy and security. Studies probing security and privacy issues in cloud infrastructures have confirmed that the distributed nature of cloud infrastructure, multi-tenant architectures that share physical resources across multiple customers, and geographic data center distribution generate complex threat environments. Cloud ecosystems experience security threats across several dimensions, such as data breaches due to poor access controls, data loss due to hardware malfunctions, traffic interception through network eavesdropping attacks, and insecure interfaces that support unauthorized access. Privacy issues in cloud computing are driven by uncertainty over the location of data and the applicability of jurisdiction of privacy legislation. Organizations that exercise retention over data sanitization before migration to cloud-based analytics platforms avoid several types of risks [10].

Table 1. *Sensitive Data Categories and Identification Techniques [3, 4].*

Data Category	Examples	Identification Method	Risk Level
Authentication Credentials	Passwords, tokens, API keys	Pattern matching, field name analysis	Critical
Financial Information	Credit cards, bank accounts	Regex patterns (16-digit sequences)	Critical
Governmental Identifiers	SSN, tax IDs, passport numbers	Regex patterns (9-digit formats)	Critical
Health Information	Medical records, diagnosis codes	Semantic analysis, context evaluation	Critical
Direct Identifiers	Names with addresses, biometrics	Automated scanning, manual review	High
Quasi-Identifiers	Age, gender, location, occupation	Combination analysis, correlation risk	Medium -High
Behavioral Data	Preferences, interaction patterns	Statistical uniqueness analysis	Medium

Table 2. *Transformation Techniques and Trade-offs [5, 6].*

Technique	Mechanism	Use Case	Privacy Level	Utility Preserved
Cryptographic Hashing	One-way hash with salt	User tracking, entity counting	High	High
Keyed Hashing	HMAC with secret keys	Session tracking, device ID	Very High	High
Partial Masking	Show first 6 and last 4 digits	Payment verification, support	Medium-High	Medium
Complete Replacement	Replace with placeholder	Passwords, security questions	Maximum	None
Generalization	Replace with broader category	Age ranges, geographic regions	Medium	High
Perturbation	Add statistical noise	Aggregate statistics, ML training	High	Medium

Table 3*. Centralized Architecture Components [7, 8].*

Component	Function	Security Property	Compliance Benefit
Event Interception Layer	Captures outbound analytics events	Non-bypassable, tamper-proof	Complete coverage of analytics traffic
Field Identification Engine	Detects sensitive data fields	Pattern recognition, semantic analysis	Systematic detection reduces errors
Transformation Rule Engine	Applies sanitization techniques	Consistent policies, audit logging	Uniform protection across integrations
Validation Module	Verifies no sensitive data remains	Fail-safe defaults, pre-transmission check	Prevents accidental exposure
Audit Trail System	Records transformation operations	Immutable logs, timestamp integrity	Demonstrates regulatory due diligence
Access Control Manager	Restricts logic modification	Least privilege, role-based access	Protects the integrity of protections

Table 4. *Organizational Benefits and Compliance Impact [9, 10].*

Benefit Category	Key Advantage	Risk Reduction
Regulatory Compliance	Prevents unauthorized data transmission	Avoids fines up to 4% revenue or €20M
Data Inventory	Centralizes visibility of external data flows	Eliminates unknown exposure vectors
Data Subject Rights	Keeps identifiers internal only	Simplifies access and deletion requests
Privacy by Design	Embeds protection at the architecture level	Prevents retrofitting costs

Third-Party Risk	Controls vendor data exposure	Mitigates analytics platform breaches
Incident Response	Limits breach scope to internal systems	Reduces breach notification requirements
Development Speed	Provides clear implementation guidelines	Eliminates security review bottlenecks
Brand Trust	Demonstrates privacy commitment	Protects reputation, reduces churn

6. Conclusions

The need to strike a balance between behavioral analytics and privacy protection characterizes a core challenge for modern software systems amid increasingly regulated contexts. Sanitization of analytics data is not just a checkbox for compliance but an inherent design tenet, making sustainable data practices that meet organizational goals and public expectations for privacy possible. The system provides systematic practices for the determination of sensitive content in event payloads, the choice of appropriate transformation methods depending on analysis needs and privacy limitations, and enforcing uniform protections on all external integrations through centralized processing designs. Organizations with sanitization functions built into core infrastructure are poised to navigate through shifting privacy environments with a continued competitive edge based on behavioral insight. Centralized architecture guarantees that protection of privacy keeps pace with organizational expansion, with new sources of data and analytics platforms supported without adding exposure vectors or necessitating piecemeal implementation efforts across disparate system elements. Transformation methods such as cryptographic hashing, partial masking, and complete substitution offer adaptable tools to support varied use cases, from making longitudinal tracking of anonymized user groups possible to removing authentication credentials of no valid analytical use. By taking a view of privacy as an architectural design necessity invested in foundational system levels instead of an operational afterthought invoked unrealistically across distributed elements, organizations establish platforms for long-term data practices. The architecture illustrates how privacy protection and valuable analysis are complementary rather than competing goals that can be fulfilled through good architectural design, systematic enforcement mechanisms, and organizational dedication to user trust as a strategic asset, versus a regulatory obligation to be minimized.

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] Arvind Narayanan and Vitaly Shmatikov, "Robust De-anonymization of Large Datasets (How to Break Anonymity of the Netflix Prize Dataset)," arXiv, 2024. [Online]. Available: https://arxiv.org/pdf/cs.CR/0610105

[2] WILLIS H. WARE, "INFORMATION SYSTEMS SECURITY A N D PRIVACY," Communications of the ACM, 1984. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/358027.358034

[3] Matthew Bach-Nutman, "Understanding The Top 10 OWASP Vulnerabilities," arXiv. [Online]. Available: https://arxiv.org/pdf/2012.09960

[4] Gloria Bondel et al., "Towards a Privacy-Enhancing Tool Based on DeIdentification Methods," Twenty-Third Pacific Asia Conference on Information Systems, 2020. [Online]. Available: https://web.archive.org/web/20230122210907id_/https://aisel.aisnet.org/cgi/viewcontent.cgi?article=1156&context=pacis2020

[5] QIANG LIU et al., "A Survey on Security Threats and Defensive Techniques of Machine Learning: A Data Driven View," IEEE Access, 2018. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8290925

[6] Oleksandr Tomashchuk et al., "A Data Utility-Driven Benchmark for De-identification Methods," Springer. [Online]. Available:

https://link.springer.com/content/pdf/10.1007/978-3-030-27813-7_5.pdf

[7] CARL E. LANDWEHR et al., "Formal Models for Computer Security," ACM, 1981. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/356850.356852

[8] LANCE J. HOFFMAN, "Computers and Privacy: A Survey," ACM, 1969. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/356546.356548

[9] Sean Sirur et al., "Are we there yet? Understanding the challenges faced in complying with the General Data Protection Regulation (GDPR)," arXiv, 2018. [Online]. Available: https://arxiv.org/pdf/1808.07338

[10] Minqi Zhou et al., "Security and Privacy in Cloud Computing: A Survey," Sixth International Conference on Semantics, 2010. [Online]. Available: https://www.researchgate.net/profile/Weining_Qian/publication/224204127_Security_and_Privacy_in_Cloud_Computing_A_Survey/links/55b6f75e08ae092e9656f9a5/Security-and-Privacy-in-Cloud-Computing-A-Survey.pdf