International Journal of Engineering Research and Modern Education (IJERME)
Impact Factor: 7.018, ISSN (Online): 2455 - 4200
(www.rdmodernresearch.com) Volume 8, Issue 1, 2023

ASSESSING DOCUMENTED DELETION CONTROLS AND AUDITABILITY ACROSS REPLICATED CLOUD STORAGE AND BACKUP SYSTEMS

Divya Vishwanath* & Paul Johnson Asamoah**
* Janardhan Rai Nagar Rajasthan Vidyapeeth, Udaipur, Rajasthan, India
** Power Life University College, Accra, Ghana, West Africa

Cite This Article: Divya Vishwanath & Paul Johnson Asamoah, "Assessing Documented Deletion Controls and Auditability Across Replicated Cloud Storage and Backup Systems", International Journal of Engineering Research and Modern Education, Volume 8, Issue 1, Page Number 93-100, 2023.

Copy Right: © IJERME, 2023 (All Rights Reserved). This is an Open Access Article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract:

This analysis examines how documented deletion controls relate to deletion auditability in replicated cloud storage and backup systems, addressing the distinction between assured deletion claims and verification evidence. A descriptive comparison uses four purposively selected published designs within a 2010 to 2022 eligibility window, with retained publications dated 2011, 2012, 2017, and 2018. Source linked records classify key management, deletion granularity, copy and version handling, and explicit deletion verification evidence. One design, representing 25 percent of the sample, explicitly describes verification evidence for associated copies. The remaining descriptions document access withdrawal, selective backup deletion, or deduplication without specifying equivalent verification evidence in the inspected material. Deletion granularity is documented for three designs. The comparison distinguishes cryptographic dependencies governing recoverability from evidence supporting assessment of deletion execution. Its contribution is an explicit separation of documented control capabilities and documented auditability, while preserving unspecified features as evidence limitations. The findings support cloud procurement and governance practices that match deletion claims to defined targets, retained dependencies, and verification coverage.

Key Words: Cloud Storage; Cryptographic Deletion; Data Auditability; Descriptive Comparison; Replicated Backups.

1. Introduction:

Deleting a cloud object does not, by itself, establish what happened to its replicas, backup versions, or decryption keys. These dependencies matter because a deletion request can target one logical object while related content remains necessary for other authorized uses. Cloud outsourcing also separates data ownership from infrastructure administration, limiting direct observation of provider operations. Jansen and Grance 2011 identify this displacement of data and services beyond organizational boundaries as a central security concern. For deletion assurance, the substantive question is therefore both what a control changes and what evidence makes that change assessable. Cryptographic deletion addresses this problem by making access depend on keys that can be revoked or destroyed. Its assurance claim differs from physical removal of every ciphertext copy. Kissel et al. 2014 define media sanitization through the infeasibility of accessing target data under a specified level of effort, making the recovery boundary central to interpretation. A design may support such inaccessibility without describing evidence through which an observer can verify the relevant operation. Distinguishing deletion controls from documented deletion auditability is consequently necessary for comparing systems whose security claims concern different objects and assumptions.

The closest research shows how key governance and deletion granularity shape those claims. Tang et al. 2012 associate outsourced files with access policies and use independent quorum key managers to support assured deletion following policy revocation. Rahumed et al. 2011 address a related dependency in version-controlled backups: selected files or versions must become inaccessible while retained versions continue to use shared content. These designs establish mechanisms for controlling access to specified targets. Their comparison also shows why selective deletion cannot be interpreted independently of the keys, policies, and retained versions on which recovery depends. Redundant storage introduces a further distinction between managing repeated content and verifying deletion operations. Lai et al. 2017 combine backup deduplication and assured deletion while limiting the key retained by each client. Dulin et al. 2018 address explicit multiple copy storage through ADM, which uses encrypted key outsourcing, a predeleting sequence, and a Merkle hash tree to associate copies and provide deletion evidence. These approaches concern related security problems but expose different information for comparison. Deduplication describes how repeated content is managed; copy association identifies relationships among deletion targets; verification evidence supports assessment of a specified operation. Their presence should therefore be recorded separately rather than treated as interchangeable indicators of security.

The reviewed evidence does not establish that a documented assured deletion mechanism necessarily includes an explicit deletion verification mechanism in its accessible description. Nor does an omitted description establish that the complete system lacks that capability. This creates a bounded measurement problem: comparisons must distinguish documented controls, documented verification evidence, and information that the inspected material does not specify. The analytical contribution is to make these distinctions explicit across selected designs. It does not depend on claiming that deletion verification is a new research topic or that the selected publications represent the complete literature. This study aims to compare documented deletion controls and deletion auditability across published replicated cloud storage and backup designs within 2010 to 2022. The central research question asks how documented auditability differs across key management arrangements, deletion targets, and copy or version handling approaches. The analysis identifies the controls described in each selected design, determines which inspected sources explicitly describe deletion verification evidence, and compares that documentation across architectures. These objectives concern observable source descriptions rather than operational deletion performance.

The evidence comprises four distinct designs published in 2011, 2012, 2017, and 2018. Each design contributes one case record supported by a source reference and extraction notes. Key management, deletion granularity, and copy or version handling is recorded as categorical attributes. Documented deletion auditability distinguishes an explicit description of verification evidence from no such description in the inspected material. This descriptive design preserves differences in technical purpose and source coverage instead of forcing unlike mechanisms into a numerical performance ranking. The completed comparison identifies explicit deletion verification evidence in one selected case, while the other inspected descriptions emphasize cryptographic access control, selective backup deletion, or redundant content management. Its contribution is to clarify the evidential boundary of those descriptions. For assurance reasoning, the comparison separates the deletion target, the dependencies governing recovery, and the scope of verification evidence. For practice, it supports more precise assessment of whether a deletion claim identifies what is affected and how the stated operation can be checked. These distinctions are relevant across cloud environments, although the four selected cases do not support conclusions about provider prevalence, national differences, or deployed system effectiveness.

2. Theory, Conceptual Framework and Research Questions Development:

This analysis draws on information asymmetry and cryptographic assurance to examine documented deletion controls in replicated cloud storage and backup systems. Data owners specify deletion requirements, while providers control the infrastructure that stores and reproduces their data. This separation limits direct observation of deletion operations. Jansen and Grance 2011 identify the displacement of data and services beyond organizational boundaries as a central cloud security concern. Within this setting, deletion auditability depends on the evidence available to assess an operation, rather than on the deletion request alone. The framework distinguishes documented deletion controls from documented deletion auditability. Deletion controls describe key management, deletion granularity, and copy and version handling. Deletion auditability identifies whether the inspected source explicitly describes evidence for verifying deletion. This distinction follows the assurance logic in Kissel et al. 2014, which defines sanitization relative to the effort required to recover target data. Cryptographic inaccessibility and physical removal therefore represent different assurance claims. Neither claim establishes that an external observer can verify deletion across every stored copy.

Key management determines who can authorize access and which cryptographic dependencies must change when deletion occurs. Revoking an essential key can make retained ciphertext inaccessible without requiring the provider to locate and overwrite each ciphertext copy. The mechanism depends on protecting the relevant keys and preventing their recovery through retained copies. Tang et al. 2012 implement policy-based deletion through cryptographic operations maintained by independent quorum key managers. Their design demonstrates how access revocation can support assured deletion, but that mechanism must remain analytically distinct from evidence that verifies a deletion operation. Deletion granularity defines the object to which a deletion decision applies. A request may target a file, a policy associated group of files, or a particular backup version. Greater selectivity can preserve legitimate access to retained content while withdrawing access to the specified target. However, selectivity also introduces dependencies when deleted and retained versions share data. Rahumed et al. 2011 address this problem through layered encryption that supports deletion of selected backup files or versions while preserving other versions that share their content. This mechanism makes the deletion target central to interpreting the assurance claim.

Copy and version handling determines how deletion interacts with redundancy. Replication supports availability, while deduplication reduces repeated storage. These arrangements create different dependencies between a logical deletion request and the data retained by the system. Lai et al. 2017 combine backup deduplication and assured deletion while limiting the key retained by each client. Their accessible description supports comparison of storage and key management arrangements, but it does not provide sufficient detail to infer deletion granularity or a deletion verification mechanism. The analysis therefore retains those limits in the case coding. Explicit copy association offers a different route to documented auditability. Dulin et al. 2018 describe ADM, which combines outsourced encrypted keys, a predeleting sequence, and a Merkle hash tree to

associate copies and provide evidence for deletion operations. The theoretical expectation is that linking deletion targets to verifiable records can improve the observability of the specified operation. This expectation remains bounded by the copies and operations represented in the protocol. It does not establish removal of undisclosed replicas or independently retained backups.

The conceptual framework organizes the four published designs through three observed attributes: key management approach, deletion granularity, and copy and version handling. Documented deletion auditability is the observed outcome. The attributes guide a descriptive comparison of control arrangements and verification evidence. They are categorical descriptions, not latent constructs or components of a numerical control index. Their connection to auditability represents an analytical relationship to examine across cases, rather than an estimated causal effect.

The analysis addresses three research questions.

- RQ1: What key management, deletion granularity, and copy or version controls are documented in the selected designs? This question is resolved through the source extracted descriptions for each case.
- RQ2: Which selected designs explicitly describe verifiable deletion evidence? This question is resolved through the documented auditability category and its coding rule. A value of one indicates an explicit description of deletion verification evidence. A value of zero indicates that such evidence is not described in the inspected material.
- RQ3: How does documented deletion auditability differ across the selected control designs? This question is resolved through a comparison of the three control attributes against the auditability category. Interpretation remains at the level of the four selected designs. The comparison preserves the distinction between an undocumented capability and an absent capability.

3. Data:

This analysis uses source documented descriptions of four cloud storage and backup designs to compare deletion controls and deletion auditability.The dataset preserves case identifiers, extraction evidence, and source links to support descriptive comparisons.

3.1 Data Sources, Population, and Sample Construction:

The evidence consists of published system descriptions rather than operational records supplied by cloud providers. The four selected designs are Fade Version, FADE, the backup system developed by Lai and colleagues, and ADM. Their publication years are 2011, 2012, 2017, and 2018, respectively. Each design contributes one observation. The resulting structure is a cross section of published designs, with publication year serving as a bibliographic attribute rather than a repeated measurement date. The source register identifies each publication through its DOI and records the inspected access route. Institutional publication records from the Chinese University of Hong Kong support the Fade Version and FADE cases. A Jinan University institutional abstract supports the Lai case, with Springer proceedings metadata providing bibliographic information. The Springer publisher abstract supports ADM. The extraction evidence consists of accessible abstracts and, for Fade Version, an inspected manuscript introduction. The dataset does not reproduce complete articles or inherit a uniform source licence.

The target population comprises published deletion designs for replicated cloud storage and cloud backup systems within 2010 to 2022. The accessible sampling frame contains four purposively selected designs with descriptions sufficient for the recorded comparison. Eligibility requires a cloud backup or explicit multiple copy setting, a deletion mechanism, and accessible primary source descriptions. General single copy deletion and device only systems fall outside these boundaries. The workbook records these selection rules but does not establish when they were adopted. It therefore documents a purposive comparison rather than a comprehensive literature census. The retained cases cover distinct technical arrangements. Rahumed et al. 2011 describe selective deletion of backup versions and files while preserving shared content required by retained versions. Tang et al. 2012 describe policy associated file deletion governed by independent quorum key managers. The Chinese University of Hong Kong Lai et al. 2017 contribute a backup design combining deduplication and assured deletion. Dulin et al. 2018 contribute a design that associates multiple copies and describes verifiable deletion evidence. The sample construction table reports verified case counts. The broader population and the number of publications screened before case selection are not recorded.

Stage	Units	Observations	Reason or Rule
Accessible frame recorded in the workbook	4 designs	4 case records	Cloud backup or explicit multiple copy deletion designs with accessible descriptions
Selected case records	4 designs	4 case records	One record for each selected design
Recorded dataset after consolidation	4 designs	4 case records	Distinct identifiers and source references retained
Final descriptive sample	4 designs	4 case records	All recorded cases enter the categorical comparison

The workbook represents the FADE publication from 2010 through its expanded 2012 record. The earlier publication is therefore not counted as an additional design. Tang et al. 2010 provide the original description of the policy-based overlay approach. The Chinese University of Hong Kong Fade Version remains a separate case because it addresses version-controlled backup deletion, although its relationship to FADE limits assumptions of case independence. The empirical setting is cloud storage and backup architecture. The cases do not represent a geographical sample, provider market, or population of deployed services. Publication coverage falls within the stated eligibility window but does not supply observations for every year. The records contain design attributes rather than flows, stocks, or period averages.

3.2 Data Integration, Cleaning, and Quality Assurance:

The workbook organizes the evidence through linked case, source, and extraction records. Case identifiers D01 through D04 distinguish the designs. Source identifiers S01 through S04 connect each case to its bibliographic reference and inspected source. The integration relationship is one selected design to one principal source record. Extraction notes preserve the basis for the recorded descriptions. Inspection of the case table confirms four unique case identifiers and four populated source identifiers. Each selected case has an extraction evidence record. No repeated case identifier appears in the final table. The documentation does not contain a screening log from which counts of earlier unmatched or conflicting records could be reconstructed.

The dataset preserves substantive descriptions rather than forcing unlike mechanisms into a common numerical scale. Key management distinguishes layered version keys, quorum governed policy revocation, a retained client key, and outsourced encrypted keys. Copy and version handling distinguishes shared backup content, cloud overlay storage, deduplication, and explicit copy association. These differences remain visible in the categorical fields. Missing information is represented through explicit documentation categories. Deletion granularity is not detailed for the Lai design. Its key management field records the reported client key arrangement while retaining the qualification that the abstract does not detail the deletion mechanism. The FADE record similarly states that replica enumeration is not described. These qualifications identify limits in the inspected evidence rather than absent system capabilities.

3.3 Variable Measurement, Construction, and Validation:

Documented deletion controls are operationalized through three observed categorical attributes: key management approach, deletion granularity, and copy and version handling. Documented deletion auditability is recorded as a descriptive category and a binary documentation flag. The measures describe what the inspected sources disclose. They do not estimate deletion speed, recovery probability, or physical erasure across hidden copies.

Variable or Indicator	Analytical Role	Definition and Construction	Source and Identifier	Unit and Frequency	Transformation and Direction	Missing Value Rule	Coverage and Limitation
Key management approach	Control attribute	Recorded arrangement governing keys relevant to deletion	Case descriptions, S01 to S04	Nominal category per design	No numerical ordering	Preserve qualifications in source descriptions	Four records; deletion mechanism only partly described for D03
Deletion granularity	Control attribute	Recorded object targeted by deletion, such as files, versions, or associated copies	Case descriptions, S01 to S04	Nominal category per design	No numerical ordering	Retain the category not detailed	Three informative descriptions; D03 remains unspecified
Copy and version handling	Control attribute	Recorded treatment of shared content, redundant storage, or associated copies	Case descriptions, S01 to S04	Nominal category per design	No numerical ordering	Preserve limits on copy coverage	Four descriptions; replica enumeration is not described for D02
Documented deletion auditability	Outcome	Whether the inspected evidence explicitly describes deletion verification evidence	Auditability field and extraction notes, D01 to D04	Categorical status per design	Distinguishes explicit documentation from no description	No description remains a documentation category	Four records; does not measure operational deletion success
Auditability documentation code	Numerical representation of outcome	One for explicit deletion verification evidence; zero when not described in inspected evidence	Outcome coding, D01 to D04	Binary flag per design	Higher value means explicit documentation	Zero never denotes proven absence of capability	Four codes, including one positive documentation record
Publication year	Bibliographic descriptor	Year of the selected publication	Source register, S01 to S04	Calendar year per publication	No transformation	No missing year in retained records	Four publication dates within the eligibility window

Variable or Indicator	Analytical Role	Definition and Construction	Source and Identifier	Unit and Frequency	Transformation and Direction	Missing Value Rule	Coverage and Limitation
Case and source identifiers	Provenance fields	Unique labels linking designs, references, and extraction evidence	D01 to D04 and S01 to S04	Identifier per record	No transformation	Every case requires a source identifier	Four linked case records

Measurement checks focus on identifier uniqueness, source linkage, and agreement between the recorded auditability category and its numerical code. The evidence notes explain why ADM receives a value of one and why the other cases retain a value of zero under the documentation rule. The workbook does not report independent recoding or agreement between multiple coders. Descriptive coverage comprises four cases for key management, copy and version handling, and auditability coding. Deletion granularity has three informative descriptions and one explicitly unspecified category. Explicit deletion verification evidence is documented for one of four cases, equivalent to 25 percent of the selected sample. The remaining three cases account for 75 percent and indicate no such description in the inspected material. These proportions describe the selected records only. The nominal attributes have no meaningful mean, standard deviation, or median. Their interpretation rests on case descriptions and category counts. The numerical auditability flag summarizes documentation prevalence without converting the control attributes into a composite index.

3.4 Analytical Framework, Identification, and Reproducibility:

The analysis is descriptive and addresses three questions: which deletion controls are documented, which designs explicitly describe deletion verification evidence, and how auditability documentation differs across the selected arrangements. The analytical procedure compares the recorded key management, deletion granularity, and copy and version descriptions against the auditability category. Each design receives equal representation as one case. The reported auditability proportion uses all four retained records as its denominator. Interpretation depends on the distinction between explicit verification evidence and the absence of such a description in the inspected text. The comparison does not identify causal effects or estimate the performance of deployed cloud systems.

Source accessibility and reporting depth constrain the comparison. A short abstract may omit a capability discussed elsewhere in the complete publication. Conversely, a protocol description may establish verification within its stated assumptions without demonstrating deletion across undisclosed infrastructure. Documentation differences therefore cannot establish a ranking of operational security. The selected designs also differ in purpose and technical ancestry. Their publication dates, backup functions, and copy handling arrangements do not constitute controlled variation. The analysis consequently interprets patterns within the four cases and preserves the source specific qualifications attached to each record. Reproducibility rests on the workbook's case table, variable dictionary, source register, and extraction evidence. Case identifiers connect each observation to its supporting description, while DOI links identify the underlying publications. The coding rule remains consistent across cases: explicit deletion verification evidence receives one, and no description in the inspected material receives zero. This alignment keeps the analytical claims within the scope of the constructed evidence.

4. Methods:

4.1 Research Design and Comparative Logic:

This analysis uses a descriptive comparison of four published cloud storage and backup designs. The unit of analysis is a distinct system design, represented by one source linked case record. The analytical objective is to identify documented deletion controls, determine which inspected descriptions explicitly provide deletion verification evidence, and compare that documentation across architectural arrangements. The comparison uses variation in how designs govern cryptographic access, define deletion targets, and handle shared content or associated copies. These attributes organize the interpretation of documented auditability. Publication year identifies the selected source version rather than an observation period for system performance. Case selection is purposive. The sample construction table records the accessible cases and eligibility boundaries. The resulting evidence supports comparisons among selected designs rather than estimates for a population of cloud services. Differences in technical purpose, source detail, and publication format remain competing explanations for differences in documentation. The analysis therefore interprets recorded descriptions without attributing causal effects to architectural choices.

4.2 Analytical Structure and Coding Rules:

The analytical structure consists of three categorical control attributes and one documentation outcome. The variable dictionary defines key management approach, deletion granularity, and copy and version handling. Each attribute preserves the terminology and qualifications needed to distinguish the selected architectures. Categories have no assumed numerical ordering. Documented deletion auditability records whether the inspected material explicitly describes evidence for verifying a deletion operation. Its numerical representation assigns one when such evidence is described and zero when it is not described. Zero therefore indicates a documentation status rather than demonstrated absence of a system capability.

Coding distinguishes deletion mechanisms from verification mechanisms. A description of key revocation, selective access withdrawal, deduplication, or prototype evaluation does not independently satisfy the explicit verification criterion. The positive category requires a description connecting evidence to assessment of the deletion operation. This rule prevents general security claims from being treated as equivalent to documented verification. The comparison retains source qualifications. An unspecified deletion target remains unspecified, and an incomplete mechanism description remains qualified. These categories preserve the boundary between available evidence and inferred capability. No composite score combines the architectural attributes.

4.3 Data Processing and Descriptive Procedures:

The analytical sequence begins with the populated case table. Case identifiers connect each design to its extraction evidence, while source identifiers connect the record to its bibliographic reference and inspected access route. Inspection checks identifier uniqueness, source linkage, and the correspondence between the auditability description and its numerical code. All four case records enter the descriptive comparison. The analysis first compares the three control attributes across cases. It then counts records with explicit deletion verification evidence and records without that description. Each count is divided by the total number of retained cases and expressed as a percentage. Every case contributes equally to these summaries.

Coverage is assessed separately from field completion. A populated field containing an unspecified category does not count as an informative technical description. This distinction applies particularly to deletion granularity and to qualifications concerning replica coverage or deletion mechanisms. Missing technical detail is neither imputed nor converted into an assumed capability. Records remain in the comparison when a source does not specify an attribute. The descriptive outputs consequently retain a common sample while making differences in documentation coverage visible.

4.4 Credibility, Replicability, and Interpretive Boundaries:

Credibility rests on traceability between case descriptions, coding decisions, and inspected evidence. The workbook preserves the case table, variable dictionary, source register, and extraction notes. These materials allow a reader to identify the source underlying each record and examine whether its description supports the assigned documentation category. Verification checks establish consistency within the constructed dataset. They do not establish that accessible descriptions exhaust the capabilities documented in complete publications. Likewise, agreement between an extraction note and its code does not demonstrate operational deletion success.

The principal interpretive boundary concerns the distinction between documentation and implementation. Explicit verification evidence describes a capability within the source's stated scope. It does not establish removal of undisclosed replicas or independently retained backups. Conversely, an omitted verification description cannot establish technical absence. The method answers the research questions through categorical comparison and transparent counting. Its contribution is a consistent separation of documented deletion arrangements from documented verification evidence, supported by records that preserve source specific qualifications.

5. Findings:

This analysis compares documented deletion controls and deletion auditability across four published cloud storage and backup designs. The findings address control variation, explicit verification evidence, and differences in auditability across the selected arrangements. Interpretation concerns the inspected source descriptions and the capabilities they document. The analytical sample contains four designs: Fade Version, published in 2011; FADE, published in 2012; the backup system developed by Lai and colleagues, published in 2017; and ADM, published in 2018. Each design contributes one case record. All four records contain descriptions of key management and copy or version handling. Deletion granularity is specified for three designs, representing 75 percent of the sample, and remains unspecified in the inspected description of the Lai design. The auditability field is populated for every case through a documentation rule that distinguishes explicit verification evidence from no such description. These records form a comparison of published architectures rather than repeated observations of operational systems.

For RQ1, the findings identify distinct arrangements for governing deletion. Fade Version uses layered encryption with version keys and targets individual backup versions or files. Its treatment of shared content preserves access through retained versions, making version dependencies central to the deletion arrangement. This description accords with the selective backup deletion mechanism reported by Rahumed et al. 2011. FADE instead associates files with access policies and distributes cryptographic operations across independent quorum key managers, as described by Tang et al. 2012. The Lai design records one retained client key and backup deduplication, but its inspected abstract does not specify deletion granularity or detail the deletion mechanism. Lai et al. 2017 therefore contribute an informative description of key retention and redundant content handling without supporting an equivalent comparison of deletion targets. ADM combines outsourced encrypted keys with explicit association among copies. Together, these observations resolve RQ1 by showing that the selected designs organize deletion around different cryptographic dependencies and different storage objects.

For RQ2, one of the four designs explicitly describes verifiable deletion evidence. ADM receives a documentation code of one, accounting for 25 percent of the sample. Its inspected description links a predeleting sequence and a Merkle hash tree to evidence for deletion operations involving associated copies. Dulin et al. 2018 explicitly present this verification function alongside encrypted key outsourcing. Fade Version, FADE, and the Lai design receive codes of zero, together accounting for 75 percent. These values indicate that deletion verification evidence is not described in the inspected material. They do not establish that the complete systems lack verification capabilities. The empirical result is therefore a difference in documented auditability, with explicit evidence concentrated in one selected case.

For RQ3, the comparison shows that selective deletion and explicit deletion verification are distinct documented properties. Fade Version and FADE specify deletion targets and cryptographic control arrangements without describing a deletion proof in the inspected evidence. ADM describes both copy association and verification evidence. The Lai record documents deduplication but provides less detail about the deletion target. This pattern indicates that descriptions of key control, selective access withdrawal, or redundant content management do not alone establish documented verification of a deletion operation. The distinction is consistent with the assurance boundary in Kissel et al. 2014, which defines sanitization through the infeasibility of recovering target data under a specified level of effort. Within the selected cases, copy association accompanies explicit verification evidence, but a single case cannot establish a general relationship between architecture and auditability. The three research questions are resolved through categorical comparison: the designs differ in their documented controls, only ADM explicitly describes deletion verification evidence in the inspected material, and auditability documentation does not follow automatically from a stated assured deletion mechanism.

6. Discussion:

A documented deletion mechanism does not automatically provide documented evidence that deletion occurred. This analysis makes that distinction visible across four cloud storage and backup designs. Explicit deletion verification evidence appears in one selected case, while the other inspected descriptions emphasize access withdrawal, selective deletion, or redundant content management. The contribution lies in comparing control arrangements separately from their documented verification properties. This separation sharpens the interpretation of assured deletion claims. Tang et al. 2012 describe policy governed cryptographic deletion, but the inspected description does not establish an equivalent verification function. The comparison therefore distinguishes what a mechanism is designed to accomplish from the evidence available to assess its execution.

The architectural differences explain why deletion assurance requires attention to dependencies. Fade Version connects deletion decisions to version keys while preserving content required by retained versions. Rahumed et al. 2011 thus demonstrate that deleting a logical backup version need not require destroying every shared data object. The analytical implication is that the deletion target must be defined before its auditability can be evaluated. Evidence about withdrawing access to one version cannot automatically establish removal of shared content. This interpretation refines the framework by treating deletion granularity as part of the assurance boundary rather than as a simple indicator of stronger control.

Copy association provides a further distinction between managing redundancy and documenting deletion operations. ADM explicitly connects associated copies with verification evidence through a predeleting sequence and a Merkle hash tree, as described by Dulin et al. 2018. Its position in the comparison suggests that identifying the objects covered by a deletion operation is relevant to interpreting the resulting evidence. However, the single documented verification case does not establish that copy association is a dominant mechanism or a sufficient condition for auditability. The contribution is a more precise comparison of target identification and verification, without converting their coexistence into a causal claim.

Reporting depth also shapes the conclusions that a comparison can support. The Lai design documents backup deduplication and a retained client key, but its inspected abstract provides less detail about deletion granularity and verification. Lai et al. 2017 therefore illustrate how an informative description of storage efficiency can remain insufficient for assessing another property. This distinction changes how undocumented features should be interpreted: limited disclosure cannot serve as evidence of technical absence. For researchers and evaluators, the practical implication is to record documentation coverage alongside capability descriptions. Doing so prevents differences in source detail from becoming unsupported rankings of system security.

The international relevance rests on architectural questions shared across cloud environments, rather than on differences between countries. The comparison directs attention to three matters that users and providers can specify more clearly: the deletion target, the cryptographic dependencies governing access, and the objects covered by verification evidence. These distinctions are consistent with the recovery effort boundary articulated by Kissel et al. 2014, while adding a separate focus on documented observability. Decision makers assessing deletion claims should therefore seek evidence matched to the stated target and coverage. The resulting theoretical refinement separates deletion control, recovery resistance, and verification scope, enabling more precise comparisons without treating them as interchangeable forms of assurance.

7. Conclusion and Implications:

Cloud deletion assurance depends on what evidence establishes about the specified deletion target. Across the selected designs, arrangements governing access, retained content, and associated copies support different claims about what deletion accomplishes. Explicit verification evidence does not automatically accompany descriptions of assured deletion. The contribution is a source traceable comparison that separates documented control capabilities from documented evidence of their execution. This distinction refines assurance reasoning by requiring claims about recovery resistance and verification coverage to be assessed separately. For managers, the findings support evaluating deletion services against specified targets, retained dependencies, and evidence requirements rather than relying on general security descriptions. Procurement teams can request explanations of which copies and versions a deletion claim covers and how execution can be assessed. Governance policies can require those boundaries to remain explicit in-service documentation and internal reviews. Operational teams can preserve links between deletion requests, affected objects, and the evidence supplied for each operation. These implications apply across cloud environments because outsourced administration and redundant storage create shared assurance questions. Their practical value lies in improving the precision of decisions and accountability, while avoiding unsupported claims about complete erasure. Clearer assurance boundaries can also support more transparent communication with data owners.

The purposive comparison provides a basis for broader evidence collection. Full text assessment could determine whether capabilities omitted from accessible descriptions are documented elsewhere. Independent coding could assess the consistency of classification decisions. A larger sample could examine how reporting practices differ across architectural families without treating related designs as independent evidence. Controlled evaluations could compare documented claims with observed deletion behavior under specified recovery and retention conditions. Such extensions would clarify which distinctions remain useful across implementations and which depend on source coverage, strengthening the transferability of the framework without assuming operational effectiveness from documentation alone.

References:

1. Dulin, Zhang, Z., Tan, S., Wang, J., & Tao, X. (2018). An associated deletion scheme for multi-copy in cloud storage. In J. Vaidya & J. Li (Eds.), Algorithms and architectures for parallel processing (pp. 511-526). Springer. https://doi.org/10.1007/978-3-030-05063-4_38
2. Jansen, W., & Grance, T. (2011). Guidelines on security and privacy in public cloud computing (NIST Special Publication 800-144). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-144
3. Kissel, R., Regenscheid, A., Scholl, M., & Stine, K. (2014). Guidelines for media sanitization (NIST Special Publication 800-88, Revision 1). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-88r1
4. Lai, J., Xiong, J., Wang, C., Wu, G., & Li, Y. (2017). A secure cloud backup system with deduplication and assured deletion. In T. Okamoto, Y. Yu, M. H. Au, & Y. Li (Eds.), Provable security (pp. 74-83). Springer. https://doi.org/10.1007/978-3-319-68637-0_5
5. Rahumed, A., Chen, H. C. H., Tang, Y., Lee, P. P. C., & Lui, J. C. S. (2011). A secure cloud backup system with assured deletion and version control. In 2011 International Conference on Parallel Processing Workshops (pp. 160-167). IEEE. https://doi.org/10.1109/ICPPW.2011.17
6. Tang, Y., Lee, P. P. C., Lui, J. C. S., & Perlman, R. (2010). FADE: Secure overlay cloud storage with file assured deletion. In S. Jajodia& J. Zhou (Eds.), Security and privacy in communication networks (pp. 380-397). Springer. https://doi.org/10.1007/978-3-642-16161-2_22
7. Tang, Y., Lee, P. P. C., Lui, J. C. S., & Perlman, R. (2012). Secure overlay cloud storage with access control and assured deletion. IEEE Transactions on Dependable and Secure Computing, 9(6), 903-916. https://doi.org/10.1109/TDSC.2012.49